2003年2月1日発行（毎月1回1日発行）第22巻第2号 昭和57年12月6日第三種郵便物認可

マイコン わかる！動かせる！プログラムが組める雑誌

マイコンベーシックマガジン

マイコン BASIC Magazine

2
2003 FEB.
550 Yen

掲載プログラムがダウンロードできる！ http://www.basicmagazine.net/

特集 記録型パソコン用DVD技術を知る！

第1回 記録型DVDドライブの選び方

大学入試センター試験対策直前マスター

数学①②「BASIC言語」の予想問題と解法 第2弾

特選投稿プログラム・コーナー

Visual Basic V6.0	ジョービン
Visual Basic V6.0	Panel Room
C++Builder4	Flied Egg
HSP V2.55	レースdeぽん
HSP V2.55	ぬりえ_ANIME
PIECE	CatGet
iアプリ	四字熟語探し
WonderWitch	Asteroid
FX-890P	パズルナイン

デジカメ小特集！

Information Catch Up 今月の製品情報

パソコン認定試験対策講座

表計算ソフトウエア

好評連載講座

- MS Office VBA…コンピュータ対戦型ポーカー
- Visual Basic…プロシージャの機能：参照渡しと値渡し
- N88互換BASIC／ActiveBasic…グラフィックにチャレンジ！
- HSP…HSP/02でじゃんけんゲームを作る
- .NET…ヒットゲーム
- PIECE…トランプゲーム「ブラックジャック」

新連載「情報」授業にも役立つ！

デジタルの世界へようこそ

編集部の学校訪問記

東京都芝浦工業大学中学高等学校

京都府立大江高等学校

福岡県立戸畑中央高等学校（ひびき高等学校）

岡山県笠岡市立真鍋中学校

価値あるマシン

emachines™

www.emachines.co.jp

N1840 ※モニターは別売です。

emachines™

¥49,800

アメリカで赤丸急上昇のイーマシーンズ

日本初

くらべてみてください!プライスバリューで選べば、やっぱりイーマシーンズ。

幅広い用途で使える、バランスのとれたパフォーマンスを低価格でご提供します。

スタンダードモデル **N1840** ¥49,800

- Intel® Celeron®プロセッサ1.8GHz
- 128MB DDR SDRAM
- 40GB HDD
- CD-ROMドライブ
- USB 2.0ポート×6
- Windows®XP Home Edition 日本語版

Windows®XP Professionalを搭載した、ビジネスに最適な低価格モデルです。

コーポレートモデル **N1845** ¥59,800

- Intel® Celeron®プロセッサ1.8GHz
- 128MB DDR SDRAM
- 40GB HDD
- CD-ROMドライブ
- USB 2.0ポート×6
- Windows®XP Professional 日本語版

ブロードバンドを毎日快適に楽しみたい方にぴったりの高性能タイプです。

ハイパフォーマンスモデル **N2040** ¥59,800

- Intel® Celeron®プロセッサ2.0GHz
- 128MB DDR SDRAM
- 40GB HDD
- CD-R/RWドライブ
- USB 2.0ポート×6
- Windows®XP Home Edition 日本語版

Pentium®4プロセッサとDVDドライブの搭載で、音楽や映像が快適に楽しめます。

マルチメディアモデル **N4010** ¥94,800

- Intel® Pentium®4プロセッサ2.0GHz
- 256MB DDR SDRAM
- 100GB HDD
- CD-R/RW・DVDドライブ
- USB 2.0ポート×6
- Windows®XP Home Edition 日本語版

これが世界標準価格だ

イーマシーンズがおすすめするMicrosoft® Windows®XP

イーマシーンズは、365日安心サポート!

年中無休（土日祝含）朝6:00～深夜24:00

■1年間の部品交換、作業料金は不要です。

■1年間の無料サポートが受けられます。

※電話、Eメールはもちろん、チャットによるサポートも行っています。

万一、故障が発生した場合でも、サポート方法をお選びいただけます。

① **EURPP**:パーツ交換プログラム

② **修理ディポ**:修理品引き取りサービス

③ **認定サービスセンターへの持ち込み**

CUSTOMER e CARE Making a Difference

Microsoft、Windowsは米国Microsoft社の米国およびその他の国における登録商標です。

ご購入·お問い合わせは、九十九電機·石丸電気まで。

TSUKUMO

パソコン本店/II/III 03-3253-5599／DOS/Vパソコン館 03-3254-3999／Parts王国 03-5298-5099／キャプチャ王国 03-3251-0531
12 号 店 03-5298-5299／TSUKUMO eX. 03-5207-5599／モニタ王国 03-3251-0099／高田馬場店 03-5292-0799
下北沢店 03-5465-0899／名古屋1号店 052-263-1655／名古屋3号店 052-259-1299／名古屋7号店 052-251-0599
名 駅 店 052-561-0599／栄 店 052-238-7399／DEPO札幌駅前店 011-757-3299／なんばParts王国 06-4396-6899

ISHiMARU

本 店 03-3255-1500／駅 前 店 03-3255-1600／パソコンタワー 03-3257-1200／世田谷店 03-3483-3111
立川ビッグワン 042-548-1400／横 浜 店 045-261-2111／青葉台ビッグワン 045-988-1400／成田ビッグワン 0476-26-2500
柏ビッグワン 04-7160-7200／柏 店 04-7171-1400／久喜ビッグワン 0480-26-1400／上尾ビッグワン 048-777-1401
水戸ビッグワン 029-255-1401／つくば店 0298-51-1400／守 谷 店 0297-47-1400／新 潟 店 025-243-1400

emachines は、インターネット通販でもご購入いただけます。 **http://www.tsukumo.co.jp/**

マイコン BASIC Magazine

http://www.basicmagazine.net

2

2003 FEBRUARY

CONTENTS

SUN	MON	TUE	WED	THU	FRI	SAT
						1
2	3	4	5	6	7	8 BM発売日
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	

特別企画

記録型パソコン用DVD技術を知る! 17
第1回 記録型DVDドライブの選び方

大学入試センター試験対策直前マスター
数学①②「BASIC言語」の予想問題と解法 第2弾 28

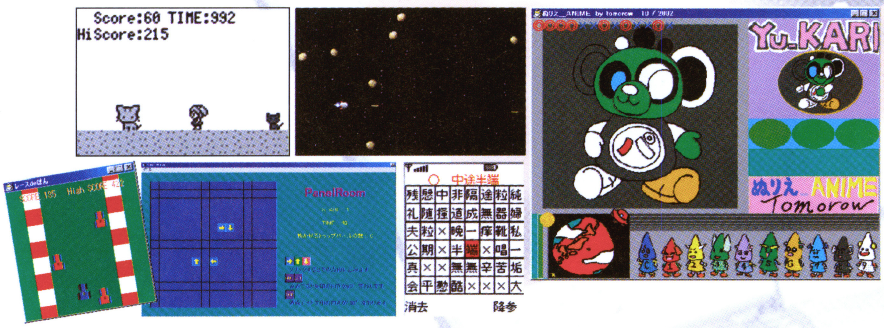
巻頭カラー企画 編集部の学校訪問記

東京都 **芝浦工業大学中学高等学校** 6
京都府立 **大江高等学校** 8
福岡県立 **戸畑中央高等学校**(ひびき高等学校) 10
岡山県笠岡市立 **真鍋中学校** 12

僕たち私たちのホームページへようこそ! 14

特選投稿プログラム・コーナー 65

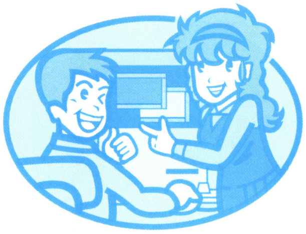
Visual Basic V6.0	ジョーピン	大石 晋也	66
Visual Basic V6.0	Panel Room	井上 秀夫	72
C++Builder4	Flied Egg	赤羊旅団	78
HSP V2.55	レースdeぽん	寺内 洋平	84
HSP V2.55	ぬりえ_ANIME	TOMORROW	87
PIECE	CatGet	宮崎麦千代	89
i アプリ	四字熟語探し	佃 貴弘	95
WonderWitch	Asteroid	江口健太郎	100
FX-890P	パズルナイン	ゆーとんパパ	107



今月のベスト・プログラマーは誰の手に!
明日のスタープログラマー 109

掲載プログラムを改造してみる!
Dr.Dのベスト・プログラムへの道 110

プログラムに関するQ&Aコーナー 112



© マイコンBASICマガジン2月号
第22巻 第2号2003年1月25日印刷
2003年2月1日発行
定価 550円(本体524円)
電波新聞社 2003 Printed in Japan
編集発行人:平山哲雄
東京都品川区東五反田1-11-15
郵便番号 141-8715
電話 東京(03)3445-6111(大代表)
振替 (東京)00150-3-51961
印刷:奥村印刷(株) 製本:(株)堅省堂

<本誌掲載の記事,プログラムの無断使用を禁止します>

デジカメ小特集!
Information Catch Up
今月の製品情報 22

マンガで学ぶ!コンピュータの世界!!
パソコンレクチャー　ザ・ビギニング 24

特別連載
パソコン認定試験対策講座
表計算ソフトウェア 35

はじめようMicrosoft OfficeでBASICプログラミング
第10回 コンピュータ対戦型ポーカーに挑戦(1) …… 43

Visual Basicアルゴリズム講座
第11回 プロシージャの機能：参照渡しと値渡し …… 52

続・ベーシックプログラミング教室
第4回 グラフィックにチャレンジ! …… 56

デジタルの世界へようこそ！
第1回 ANDとORから始めよう …… 62

PCハードウェアテクニック
誌上パソコン組み立て教室 …… 114

HSPdeネットワークゲーム！
第2回 HSP/02でゲームを作る …… 120

.NETでゲームを作ろう！
第4回 ヒットゲーム「前編」 …… 124

PC用携帯端末 PIECEプログラミング
その6　トランプゲーム「ブラックジャック」編 …… 132

ロボ道場
「OPEN−R SDK」プログラミング②　画像処理に挑戦 …… 139

電子工作にチャレンジ
『大江戸からくり人形』の製作 …… 143

読者のコーナー
あくせすROOM／プレゼント …… 147

ソフト・メーカーのPRページ
SOFTWARE MAKER HOT INFORMATION …… 148

• 3月号は2月8日発売です！•

広告索引

ハード		秋月電子通商	記事中146
ソニー	表4	HIT&RUN	後付152
アイコム	前付カラー16	HRDサービス	後付152
ソフト		パソコン工房前橋店	後付153
日本ファルコム	表3	DOS/Vアセラシステム	後付153
パソコンショップ		シグナル	後付153
九十九電機	表2〜前付カラー3	パシフィックネット	後付154

好評発売中!!

楽しく学習できる
新 まんが
電気学入門

監修：江田　稔
(文部科学省初等中等教育局
漫画：原子　力
●A5判／247頁　送料 380円
定価1,500円（税込）

電気のはたらきから
パソコンのしくみまで
わかりやすくまんがで解説！

主な内容

●第1章／静電気
静電気（まさつ電気）／正の電気と負の電気／はく（箔）検電器／静電気と電流／雷とは？ ……他

●第2章／乾電池と豆電球
電池／電池の原理／乾電池のしくみ／豆電球のしくみ／電気を通すもの、通さないもの／導線のつなぎ方／配線図と回路記号／豆電球の直列つなぎ ……他

●第3章／電流と電圧
閉回路と開回路／電流／電荷／電圧／抵抗（電気抵抗）／抵抗のいろいろ／抵抗値の見分け方／オームの法則／電流計（直流用）電圧計（直流用）のしくみ ……他

●第4章／回路計算
電圧の総和／電流の総和／電圧のかかり方／抵抗の直列接続と合成抵抗値／合成抵抗の中の未知抵抗／内部抵抗の求め方／キルヒホッフの（第1）法則 ……他

●第5章／電気エネルギー
電流による発熱作用／電流による発熱原理／発熱量の測定／ジュールの法則／抵抗の接続方法と発熱量の大きさ・電力／電力量／積算電力計（交流用） ……他

●第6章／磁石と電磁石
磁石と磁極／N極とS極／磁力の大きさ／磁気誘導／N極とS極の分離／磁力線／地磁気／電流の大きさと方位磁針の振れ／右ねじの法則 ……他

●第7章／電磁誘導
電磁誘導／レンツの法則（誘導電流の向き）／ファラデーの電磁誘導の法則／コイルの巻き数と誘導電圧／電気ブランコによる電磁誘導 ……他

●第8章／電流と電子
火花放電／真空放電とグロー放電／蛍光灯と電球／クルックス管と陰極線／原子と電子／電子と火花放電／トランジスタの構造 ……他

●第9章／パソコン
パソコンって何？／パソコンでどんなことができるの？／パソコンはどう動くの？／パソコンのいろいろな形態／パソコンの簡単な歴史／パソコンの仕組み—CPU・メモリ・補助記憶装置・フロッピーディスク・ハードディスク・CD-ROM／周辺機器—キーボード・マウス・ディスプレイ・プリンタ・スキャナ・デジタルカメラ（デジカメ）／OSの働き—OSって何？／OSの働き—OSの種類は？／日本語入力ソフト／アプリケーションソフト／LAN（ラン）て何？／USBとは？／パソコン関連の単位／インターネットとは？／インターネットはどうしてできたか？／インターネットサービスプロバイダ／インターネットで何ができるか？／WWWとは？／電子メール

電波新聞社
東京本社　電話（03）3445-8201（ダイヤルイン）
〒141-8715 東京都品川区東五反田　1-11-15 電波ビル

コンピュータ委員会のみなさん

編集部の学校訪問記

芝浦工業大学中学高等学校（鈴木渉校長）は，芝浦工業大学併設の中高一貫教育の普通科課程男子校です。前身は大正11年（1922年）設立の東京鐵道中学，2002年に創立80周年を迎えました。中学校が設置された昭和57年（1982年）に現校名となり，中高一貫教育がはじまりました。教育課程は6年間を2年区切りの3段階に分け，前期では基礎学力の育成，中期は希望と適正によって芝浦工大進学コース，他大学進学コースに分かれた教育，後期には科目選択を導入し大学進学に備えます。

コンピュータの利用も，多種多様な情報を把握・処理する能力を養うことを目的に，多くの場面で取り入れられています。今回は，情報科目の独特な授業のようすと，課外活動を中心に紹介しましょう。

東京都私立芝浦工業大学中学高等学校

http://www.shibaura-it.ac.jp/itabashi/

授業，委員会活動，部活動に見た！とにかくコンピュータが大好きな生徒が大勢！

生徒は全員ユーザーIDとパスワードを取得

同校のパソコンルームには，東芝の「EQUIUM5050」（OSはWindowsXP Pro）が60台設置されています。ディスプレイはアイ・オー・データ機器の17インチ液晶モニタ「LCD-AD17CS」です。どのPCからでもインターネット接続が可能で，専用線によって芝浦工大のサーバにつながっています。

特徴的なのが，入学と同時に生徒1人1人にユーザーIDとパスワードが発行されること。どのパソコンからでも，いつでも同じ環境で操作が可能になっています。こうした環境が，生徒自身のモラルと自己管理の徹底にもつながっています。

このパソコンルームは，管理の須田雄一先生が常駐しており，毎日昼休みと放課後をオープンルームとし，生徒は自由に利用することができます。

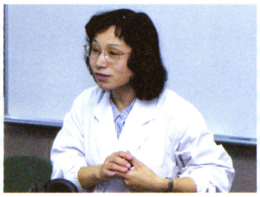
いち早く取り組んでいる「情報」科目

パソコンルームは同時に二つの授業ができるように，真ん中に間仕切りが開閉する仕組みになっています（先生機も2台あり，部屋のPC配置が左右対称になっている）。取材時には「情報」科目の二つの授業を拝見しました。神長京子先生担当の高校1年生の「レゴロボットの授業」と西元進哉先生担当の中学3年生「プレゼンテーション」の授業です。

同校では，企業などの情報教育モニター制度に積極的に応募し，当選した機材を授業などに生かしています。「レゴロボットの授業」で使用しているLEGO dactaの「教育用レゴマインドストーム」もその一つです。

授業では2人一組になって「レールからそれないように8の字を描き，壁にタッチしたらバックし，音を鳴らす」と

◀パソコンルームに設置の東芝「EQUIUM5050」。生徒個人のID，パスワードが発行されていて，どこからでも自分のパソコンになる



◀案内をしていただいた神長京子先生。中学技術科、高校情報科担当、パソコンルームの管理もしている

◀プレゼンテーションの授業のようす

コンピュータ委員会の活躍

同校では，生徒会各委員会の一つに「コンピュータ委員会」を設けています。現在は中学3年生を中心にした約80人が在籍しています。

主な仕事はパソコンルームの管理。オープンルーム時は委員が身分証明書と引き換えに使用するPCを指示します。終了時も最後に点検していきます。

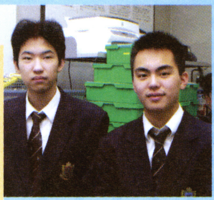
そのほか，小学生とその保護者を対象にした体験学習（毎年4回実施，各回60組が訪れる）では，カレンダー作成の実習指導，芝生祭ではインターネット，動画作成，名刺作成，カレンダー作成，レゴロボット製作などを行っています。

委員会の新聞「PCインフォメーション」の発行も行っています。

毎年夏には猪苗代合宿が行われ，タイピング練習，情報処理活用能力検定（J検）試験対策などPC活用能力を高めます。また会津大学を見学，モーションキャプチャシステムなど最新システムを体験します。同校は平成10年（1998年）に校外学習を行う場として猪苗代教育センターを開校しました。25台のPCが設置されるなど，さまざまな学習に対応しています。

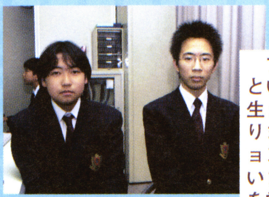
また，委員たちのがんばりに「同窓会奨励賞」も受賞しました。

4代目現コンピュータ委員長の斉藤雄一郎くん（高校2年生，右）と3代目元委員長の藤沼暢幸くん（高校3年生）



オープンルームはインターネットや授業の課題制作で使用する人が多いです。委員会は各クラスのコンピュータが好きな人たちで構成されていて，僕も小学校5年生からはじめて，現在パソコンの面白さを追求しています

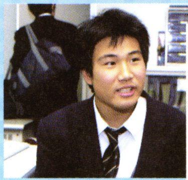
「受付け管理システム」を作成した芦川滋くん（高校3年生，右）とデバッグ作業担当の林洋平くん（中学3年生）



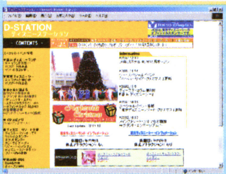
プログラミングが好きということもあって，先生から頼まれてVBで作りました。何度もバージョンアップを繰り返しています。これからC言語を学んでいきたいです

▲オープンルーム時は，身分証明書と引き換えに使用するPCを与えるのも委員会の仕事。使用状況は「受付管理システム」で一目瞭然

個人のホームページ「D-STATION ディズニーステーション」を制作。おすすめ5000本や中学生新聞でも紹介されたという木崎秀安くん（高校1年生）



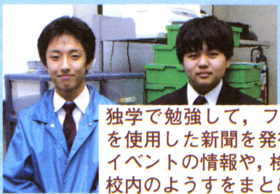
中学2年生になった頃ホームページ作りに興味を持ちました。好きなディズニーランドについてのガイド情報を少しずつ作り上げていったら，注目を集めるようになったんです。励みになります



http://www.d-sta.com/

委員会の新聞「PCインフォメーション」。2001年4月から発刊を開始し，2002年度は3回発行した

「PCインフォメーション」の制作に取り組む白鳥卓くん（中学2年生，右）と後藤星来くん(中学3年生)



独学で勉強して，フリーソフトの「朝刊太郎」を使用した新聞を発行しています。芝生祭などイベントの情報や，検定試験情報などとともに，校内のようすをまとめています

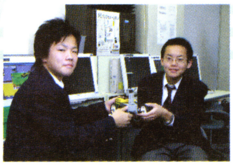
いう課題に取り組んでいました。タイヤやギアの大きさを変えたり，ブロックの組みかたやプログラムを工夫したり，それぞれ思い思いのロボットを製作していました。

神長先生も「結果が一つではなく，さまざまなアイデアが湧き出てきます。生徒は予想どおりに動くか確認しながら試行錯誤しています」と，レゴロボット導入2年目の取り組みに成果を見出していました。最終的には，結果をパソコンでレポートとしてまとめるそうです。

同校では中学3年生を対象に毎年「海外教育旅行」が組まれています。「プレゼンテーション」の授業では，9月はじめから2週間アメリカにホームステイで学習してきた内容を各人がレポートにまとめ発表するというものでした。

PowerPointを駆使し，それぞれ滞在したホストファミリーや名所の紹介などを，堂々と（中には面白おかしく）発表していました。

◀レゴロボットの授業のようす



マイコン同好会

理科部から独立して活動中の「マイコン同好会」は現在中学生中心の30人で構成されています。理科部と同じ化学実験室の片隅で，学校の払い下げPC，各人が家から持ち込んだPC，秋葉原で購入したPCなどを駆使し，ゲーム・プログラミングとホームページ制作中心の活動を続けています。最近では仕組みや構造の勉強のために，パソコンの解体も行っています。

また理科部や写真部と合同の夏合宿も恒例行事で，PCスキルアップのほか，芝生祭発表用の作品制作を行います。

◀家ではMac，学校ではWindowsと両刀使いの会長・村上慎弥くん（中学3年生，右から2人目）と理科部の面々。左から瀧井数馬くん（中学2年生），高橋慶伍くん（高校2年生），小川明雄くん（中学2年生）

小学校1年生からMacを使ってました。今後の同好会活動はホームページ上で，オリジナルゲームや絵などのダウンロードコーナー，HTMLやプログラム解説などをやっていきたいです。

▲顧問の川野一忠先生（化学担当）。芝浦工大の講師も兼務

◀活動場所は化学実験室の一角。1台のみ電話線を引いてきてインターネットに接続できる。ゲームプログラムに夢中の生徒のみなさん

◀過去には，旺文社主催（文部科学省後援）の「全国中学校・高校生パソコンソフトコンテスト」で特別賞やアイデア賞を受賞した実績がある

◀Visual Basicの勉強中という増田純太くん（中学2年生）。ベーマガでBASICをマスターしたという

◀本棚にはベーマガのバックナンバーがズラリ！図書室からの払い下げだそうだ

芝浦工業大学中学高等学校　〒174-8524　東京都板橋区坂下2-2-1　☎03-5994-0721

編集部の学校訪問記

マイタウンマップコンクールに取り組む2年生有志のみなさんと勝井真太郎先生

大江町は酒天童子（しゅてんどうじ）などの鬼伝説をテーマとした町づくりをしているユニークな町です。京都府立大江高等学校（塩見裕校長）でも鬼は生徒に親しまれ，授業の中にもテーマとして取り入れられるほどです。

同校は明治41年（1908年）に創設された加佐郡立河守蚕業学校が母体となっています。昭和60年（1985年）に農業科を廃止し，平成元年（1989年）にはソフト経済科を設置するなどの大きな変化を経て現在に至っています。普通科第Ⅰ類114人，普通科第Ⅱ類115人，ソフト経済科236人の計465人が学んでいます。

京都府立大江高等学校

http://www1.kyoto-be.ne.jp/ooe-hs/

バーチャルカンパニーの実践/各種コンクール参加で実力アップ！！

全国からも注目を集める バーチャルカンパニーの実践

大江高校では，コンピュータを使用した情報教育も，実践を主体に実績を積み上げてきています。コンピュータ環境は，4教室に160台が設置されており，恵まれた環境を活かした実践教育の中で，大きな柱となっているのが“起業家教育”です。

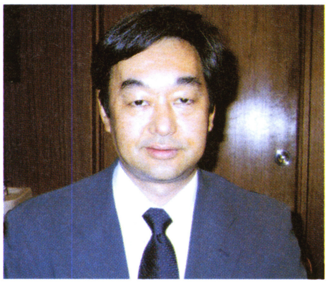
全国からも注目されている“バーチャルカンパニー（VC＝仮想企業）”を取り入れた授業は，今年度で3年目の取り組みとなります。VCとは，ネット上で架空の商品を取引する会社のことです。

VCでは，参加者は商品の売買，給料の支払い，納税など，実際の企業と同じ活動をします。また会社からもらう給料で，他のバーチャルカンパニーから商品を購入します。

今年度のバーチャルカンパニー2002は，小学校から大学まで日本全国の30校近くが参加しています。このプログラムは，世界のネットワークプログラムの一部でもあり，30か国以上，約3,000校が参加している大規模なものです。

もちろん，バーチャル（仮想現実）のため，本物の金銭のやり取りはありませんが，生徒がビジネス経営を肌で感じられる効果を生み出しています。

生徒は地元の支援企業のサポートを受けながら，会社のホームページを作って商品を売り込みます。取引のホームページは2002年11月1日に立ち上がりました。全国の参加校と取引をするため，自社の商品を買ってもらおうと生徒は真剣です。今年度，同校の生徒は，㈱The インナー，㈱オーティコカンパニー，㈱にしゆう，無名㈱という四つの会社を，1社あたり14人で作りました。前2社はグンゼ宮津工場の支援を得て下着の開発・販売をしています。あとの2社は大江町商工会の支援を得て，鬼をモチーフにした日用品・グッズなどを販売しています。



▶ソフト経済科の永山隆司先生。ＶＣ実践やコンピュータ管理などの中心となっている

支援企業からのアドバイスに 学習意欲も高まる

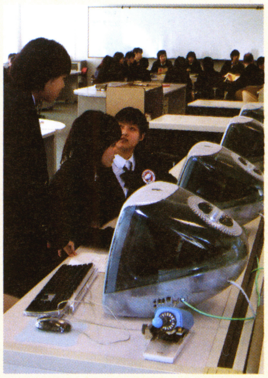
2002年11月22日には各会社のホームページの発表が支援企業の前で行われました。担当の永山隆司先生は「Power

▲2002年春に，CAV実習室のパソコンがNEW iMac40台に更新された

▲全国的にも珍しい映像スタジオを持っていて，CM制作やビデオ編集にも取り組んでいる

総合実践「バーチャルカンパニー」プレゼン授業

▲視聴覚室でのホームページとチラシを使ったプレゼンテーション。4社それぞれユニークな商品をアピール



▲総合実践室には「iMac」が8台。支援企業のアドバイスをもとにホームページを再チェック

◀プレゼンには、ノートパソコンが活躍

▲場所を総合実践室に移し，支援企業からのアドバイスをもらう。大江町商工会の公手さん（上），松本さん（左），グンゼの安藤さん（右）も一生懸命指導していた

◀VCの実践から生まれた商品。現在も駅のおみやげ屋などで販売されている

Pointを使ったプレゼンテーションの勉強の場にもなります」と話しています。

発表では，「マスコットキャラクターのオニイちゃんを“大江町のキティーちゃん”にしたい」「マニアにはたまらない商品」（にしゆう）と，自社商品を売り込むと同時に，ホームページで工夫した点などをアピールしていました。

支援企業からは「品数が多いのはいいが，メイン商品をどれにするかを決めた方がいい」「文字が小さくて読めない」「企業は地域社会，お客さんのために仕事をすることが大事」など厳しい意見も出ていました。

VCで出たアイデア商品を実際に販売するという面白い試みも実現しました。3年前のバーチャルカンパニーで反応の良かった商品を，大江町で昨年11月上旬に行われた「鬼面まつり」で販売しました。商品は，しおり（50円），香り姫（におい袋，100円），ストラップ（500円）の3商品。用意した商品は午前中で完売したものもあり，生徒からは「直にお客さんの反応が見られて良かった」と，その成果に手応えを感じていました。

これらの商品は，現在でも大江駅前のおみやげ屋さんなどでも販売されています。永山先生は「生徒たちが実際に販売することによってお客の表情が分かる」と，今後もVCで好評なものは商品化を考えていくつもり。生徒のバーチャルカンパニー熱を上げそうな試みです。

CG/ホームページなど表現の場を積極的に生かす！

同校の学校案内やPTA会報の表紙に，生徒が作成したCGが使用されているのも特徴です。「生徒に自分の表現を発表する機会を積極的に与えていく」（永山先生）という方針で取り組んでいるもので，大江町からの依頼によるポスター作りなどでも実績を上げています。

「第9回マイタウンマップ・コンクール・ぼくの街わたしの村～インターネットで見る生活地図づくり～」にも参加しています。これは，地域の魅力を伝える「生活地図」をいかにユニークな視点から捉えるかが評価基準になっています。

同校は大江町を中心に各地から生徒が集まっています。その強みを活かして大江町，福知山市，宮津市，綾部市，舞鶴市，加悦・野田川町の紹介を集合させたホームページ「CHU・丹・GO!!～わたしたちの街にズームイン～」を2年生の有志14人で作っています。

取材時は，2002年12月31日の締め切りを目前に控えた時期で，ホームページ制作の仕上げに，放課後ぞくぞくとコンピュータ室に集まってきました。

同校では，そのほか各種コンテストへの参加も積極的で，「プログラミング競技会」での優勝をはじめ，「映像マルチメディアインターハイ」でのマルチメディア部門最優秀賞受賞，「近畿高等学校総合文化祭」最優秀賞など輝かしい実績を上げています。

▲ICTラボ室には，NECのワークステーション「Express5800」（Windows2000）が40台。その他，情報処理室と文書処理室があり，FMV（Windows95）がそれぞれ40台

▲マイタウンマップコンクール参加作品「CHU・丹・GO!!」で大江町担当の河田彩さん（左）と藤田静香さん

▲修学旅行で得た情報をもとに沖縄旅行パンフレットを作成

◀2002年度「近畿高等学校総合文化祭」では、放送部門、ビデオメッセージ小部門で最優秀賞を受賞。賞状を持つ寺井夢子さん（左）と倉元綾子さん

大江高等学校　〒620-0303 京都府加佐郡大江町字金屋57　☎ 0773-56-0033

総合情報コース3年生のみなさん。1人1人話すと素直な生徒が多い

編集部の学校訪問記

福岡県立戸畑中央高等学校は，来年度（平成15年4月）「福岡県立ひびき高等学校」として生まれ変わります。名称だけでなく，学校の制度が全日制から単位制（大学の授業と同じ形式）へと変わります。

同校の前身は昭和3年（1928年）開校の戸畑実科女学校で，戸畑高等女学校を経て，昭和24年（1949年）に戸畑中央高校と改称，創立75年目を迎える今春，歴史と伝統に加え新たな息吹が芽生えるわけです。

同校は3年前から，普通科のなかに情報教育を積極的に取り入れた「総合情報コース」をスタートさせています。今回はこの総合情報コース3年生のみなさんの活躍ぶりを紹介します。

福岡県立戸畑中央高等学校
（ひびき高等学校）

http://tobatachuo.fku.ed.jp/（http://hibiki.fku.ed.jp/）

平成15年4月から単位制へと生まれ変わる！「総合情報コース」生徒の活躍ぶり

▲ホームページもリニューアル中！

環境の街・北九州市の伝統校

戸畑中央高校は製鉄所で日本の高度成長を支えた北九州市にあります。北九州はかつて公害の街としても有名でしたが，企業，住民，行政の努力で環境の街として世界的に知られるようになりました。戸畑中央高校の近くにも北九州市の環境施設があり，高校ではこれから積極的に授業で環境を取り上げていく予定です。

取材時，案内してくれたのは現在「ひびき高校準備室」にいる山原智先生です。物理が担当の山原先生は30年前プログラミングをはじめました。まだパソコンという言葉がなく，マイコンと言われていた時代で，自分でプログラムしなければソフトもなかった時代です。

▲「ひびき高等学校」になる来年度から新校舎の建設がはじまる

県内で唯一 総合情報コースを設置

戸畑中央高校に設置されている総合情報コースは，各学年1学級で，新学習指導要領の新教科「情報」を先取りした情報教育を行っています。1年次と2年次はコンピュータの基礎，プログラミング，画像処理などを学び，3年次には卒業制作として各自の個性を反映したグラフィック作品やマルチメディアソフトを制作します。

山原先生は「今日は総合情報コースに在籍する3年生の授業を二つお見せします。一つはデータベースのプログラミングで，もう一つはCGの授業です」と案内してくれました。

データベースとCGの授業を拝見

コンピュータルームで行われた井上裕美先生の授業では，まず無料で利用できるデータベースソフト「MySQL」

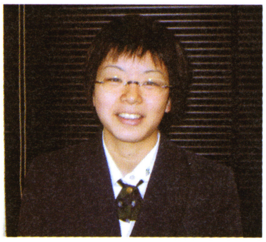
▲笑顔がとてもステキな大垣洋之校長先生。この笑顔で生徒の活躍を見守る

▲30年前からプログラミングをやっていたという山原智先生。新しい情報教育に情熱を注いでいる

3年生総合情報コースのみなさん

◀授業中にぎやかだった3人組。
一番右の上永雄馬くんはイタリアン料理人をめざしてアルバイト先で修行中。「料理を覚えるためタイムカードを押す時間前から店に行きます」エライ！
真ん中の小林大介くんは少林寺拳法2段の腕前。大会優勝経験もある強者。
一番左の平尾雄一郎くんはコンピュータ好きで，海外滞在の経験もあり国際志向です。水泳も得意

▲となりの生徒にプログラミングを教えたり秀才ぶりが目立った森冨将平くん。将来は「システムエンジニアになりたい」



▲橋本明日香さん。高校のマルチメディア部にも所属。「この3年間でアプリケーションソフトの操作は上達しました」

▲工藤亜衣子さんは将来になりたい職業は「イルカの調教師」とか

▲先生から手ほどきをうけるのは今泉博貴くん。「3年間でコンピュータは上達したと思います」知り合いの作業も手伝えるようになった

▲総合情報コース1年生を直撃！マルチメディア部副部長をつとめる大本光太郎くん。将来の夢はゲーム関係の仕事をすること。好きなゲームはナムコの「サモンナイト」とか

を利用して，料理のレシピデータベースを作っていました。「createdatebase ryouri; use ryori;……」と命令文を先生が説明していきます。生徒のみなさんのパソコンモニタには「豚汁」「オムレツ」「カツ丼」などの単語が並んでいます。料理と材料を組みあせてデータベースの作り方を学んでいきます。

続いてAdobeの「フォトショップ」を使った本田禎之先生，森とも子先生による授業です。錆びてボコボコになった青い車の映像を，さまざまな方法で修正し新車同様に変えていく様子は圧巻。生徒諸君は自然体で取り組んでいました。

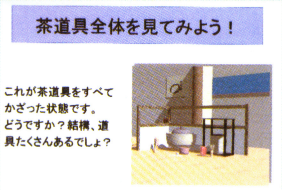
新しい学校へ 校長も熱い視線

大垣洋之校長先生はギター，三味線，胡弓など弦楽器をオールマイティに弾け，とても音楽にくわしい方です。コンピュータの音楽ソフトも使用経験を踏まえて「生徒もみんな自分を表現したくてたまらない。ただコンピュータのコミュニケーションに乗っているだけでは，安直なものになってしまいます」とお考えを話してくれました。「何をするにも結局相手にするのは人間。コンピュータは手段の一つです。自分を伝えたいとき，電話で話したり会った方がいい，あるいは紙に書く方がいい。そのときに一番良い方法を選択することが大切。私たちもそれを教えています」ということです。

戸畑中央高校は来年度生まれ変わります。「単位制になることは，生徒に自己責任・自己選択が要求されるということです。これはいわばリトル社会。さまざまなチャレンジする勇気を生徒には持ってもらいたいです」と大垣校長先生。

新しい「ひびき高等学校」で生徒はいろんな意味で「選択」ということを学んでいくのでしょう。

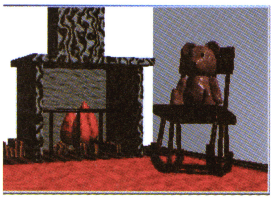
総合情報コース卒業生のCG作品

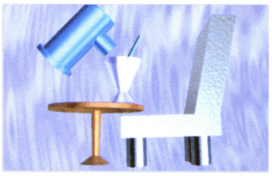


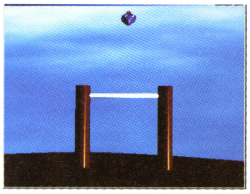
▲技術者として北九州の代表的な企業に就職した生徒が自作しCG。光と影を表現しているのはさすが

▲同じくCG作品集から。某有名キャラクターの「昔の顔」?。シュルレアリスムの傑作といえるかも

総合情報コース3年生が1学期に制作した作品群







▲じつはこの中にはアニメーションもある。動きが見せられないのが残念

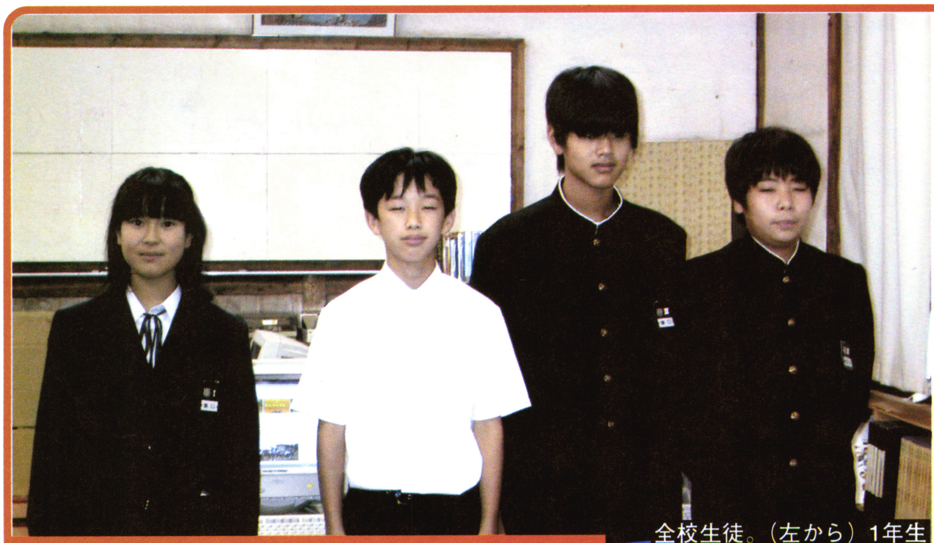
戸畑中央高等学校（ひびき高等学校）〒804-0041福岡県北九州市戸畑区天籟寺1-2-1 ☎093-881-2355

編集部の学校訪問記

岡山県笠岡市立真鍋中学校

http://www.city.kasaoka.okayama.jp/manabehp/cyu/manabe01.htm

全校生徒４人！生徒たちが積極的に活動している様子を語りかけるホームページ



全校生徒。（左から）1年生の峯口由美さん，濱西健誠くん，3年生の峯口正也くん，森本雄也くん

笠岡市立真鍋中学校（池田敏夫校長）は，カブトガニが有名な笠岡市沖の瀬戸内海に浮かぶ真鍋島にあります。真鍋島は笠岡港から高速艇で44分，普通船で1時間10分のところに位置する，菊の栽培が盛んで自然豊かな島です。

同校は，1947年（昭和22年）開校。全校生徒は，現在，3年生2人，1年生2人の合わせて4人という，とても小さな学校ですが，2001年（平成13年）に行われた財団法人国際コミュニケーション基金（ICF）主催「第4回ICFホームページコンテスト」で「パソコン部のコーナー」が団体の部で最優秀賞受賞の実績を残すなど，コンピュータ教育も活発に，そして地道に行われています。

▲学校の様子，真鍋島の紹介，いろんなコンテンツが目白押し！

◀生徒の生き生きとした活動を温かい目で見守る原範幸教頭先生

納得がいくまで一生懸命キーボードタイプレッスン！

同校では，一つの教室を図書室とパソコン室にわけて使っており，パソコン室には6台のパソコンが設置されています。富士通の「FMV-6000CL」（OSはWindowsXP）が4台あり，この4台はLANが組まれています。その他，コンパックの「PRESARIO 4130」，NECの「VALUESTAR U」（PC-VU50L25D）がそれぞれ1台ずつという状況です。

取材当日は，このパソコン室で行われた1年生の技術の授業を拝見しました。1年生の峯口由美さん，濱西健誠くんは授業で初めてパソコンについて勉強するそうで，パソコンの構成や名称，フロッピーディスク，CD-ROMの役割などを学んでいました。

授業を行う原範幸教頭先生は，自分が300万画素のデジカメを買ったときの話や，パソコンを買ったとき店員から聞いた「パソコンを調理場に例えると，CPUはコックの腕前，メモリは調理場の広さ，ハードディスクなど補助記憶装置が冷蔵庫で，それらのバランスが大事」という話も披露。パソコンなど実物を使っての説明，身近の出来事や例えを交えたわかりやすい話に，生徒も積極的に授業に参加していました。

▲懐かしさを感じさせる木造校舎

その後，「ハイパーキューブNet」（スズキ教育ソフト）のキーボードレッスンでブラインドタッチを行いました。レッスンの最後に表示される“間違えたキー”とその数に一喜一憂しながら，間違えなくなるまで繰り返して夢中で練習していました。

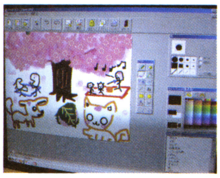
HPコンテストで最優秀賞！活動の場を広げるパソコン部

中学生と高校生を対象に「自分発，

▲「パソコン部のコーナー」。3択クイズによる同校の紹介もさらにバージョンアップ！ついつい引き込まれてしまう作りが圧巻

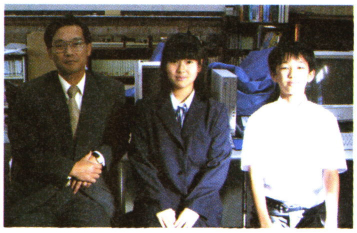
▲今年1人だけのパソコン部・峯口さんの食べ物をテーマにした2003年カレンダー

▲パソコン部の亀山紀子先生と記念に

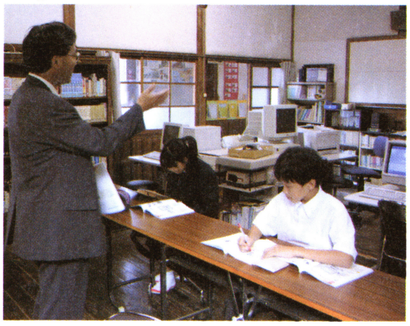


▲2003年カレンダー，3・4月「さくらもち」の絵を制作中

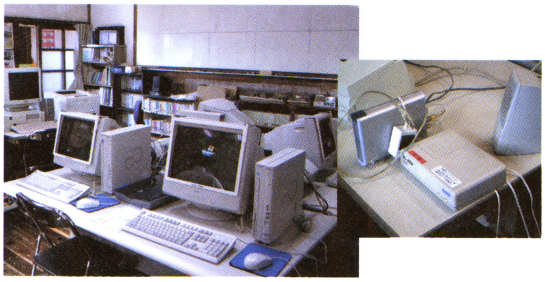
▲パソコン部の活動の様子。2003年のカレンダーを制作中。完成品は同校ホームページにアップされている



▶授業が終わってハイ！ポーズ



▲原教頭先生の1年技術の授業中



▲同校のインターネット回線はISDN。地域によっては，HL-ADPなどのブースタを設置しないとデジタル回線が開通しない場合もある。手前の4台がLANでつながっている

新コミュニケーション」をテーマに，自己表現をとおしてインターネット時代のコミュニケーションを考えることを目的として毎年開催される「ICFホームページコンテスト」。2001年，その第4回開催で最優秀賞を受賞したのが，同校パソコン部です。

同校は，インターネットが導入された1998年（平成10年）に学校のホームページを作成，その年にパソコン部も“復活”しました。というのは，それ以前にもパソコン部はあったのですが，指導していた先生が異動されたため，一時中断していたようです。

パソコン部のホームページは，毎年それまでのホームページを下地に制作しています。パソコン部を指導する亀山紀子先生も「コンテストで最優秀賞を受賞したのは，その年の“パソコン部のコーナー”だけでなく，毎年の積み重ねが評価されたのだと思う」と，何年間にもわたる地道な努力が評価されたことに喜びもひとしおのようです。

最優秀賞を受賞したときのパソコン部・山下祐紀さんが昨年卒業し，今は1年生の峯口由美さんがただ1人の部員です。テスト前以外は，毎日放課後1時間ぐらい活動を行っています。活動の成果はホームページに残すようにしているのも特徴で，歴代の生徒たちも制作物が全世界に発信される喜びで，毎日がんばっていたようです。

絵を描くのが好きな峯口さん。すでに小学校1年生くらいから学校のパソコンで「ハイパーキューブ」（スズキ教育ソフト）を使って絵を描いていたそうです。「絵を描いたり，ホームページが作れたり，パソコンっておもしろそうだなと思ったから」というのが入部した動機だそうです。

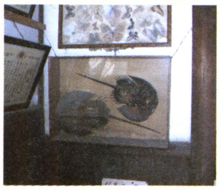
当日は，ホームページに載せるための2003年カレンダーを作っていました。食べ物をテーマにした絵を「一太郎スマイル」で描いているところで，3・4月のさくらもち，11・12月のケーキをテーマにした絵を仕上げていました。その他，1・2月がみかん，7・8月がかき氷，9・10月が焼きいもだそうで，5・6月がまだ決まっていないということでした（その後かしわもちに決定。カレンダーはホームページに掲載中）。

パソコンで絵を描くとき，「細かいところが難しい」と峯口さん。画面を部分的に大きくしたり，全体を表示したりとうまく切り替えながら，取り組んでいました。

なお，ホームページの制作にはIBMの「ホームページビルダー」が使用されています。

真鍋島を調べよう

▼真鍋島から帰る船で見た夕暮れ



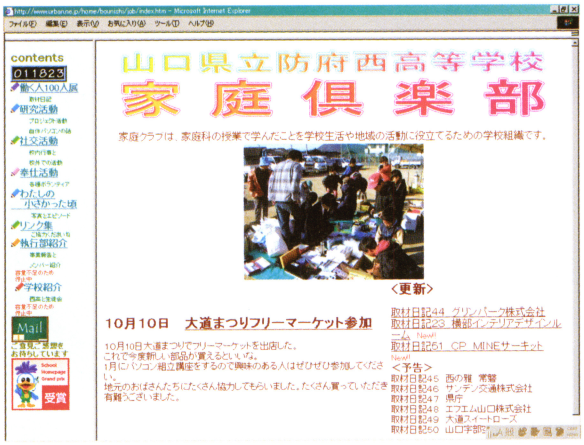
▲カブトガニが有名な笠岡市の学校らしく校内には標本が…

真鍋中学校　〒714-0037 岡山県笠岡市真鍋島4240番地　☎ 0865-68-3609

全国の学校Webを紹介

僕たち私たちのホームページへようこそ！

今回はさまざまなホームページコンテストでの受賞歴をもつ実力派を厳選しました。



山口県立防府西高等学校家庭クラブ「働く人100人展」

http://www.urban.ne.jp/home/bounishi/job/index.htm

2001年度「第8回マイタウンマップ・コンクール」厚生労働大臣賞受賞作品。学校全体のサイトではありませんが，クラブ単位でコンテスト·エントリーしたサイトが閲覧できます。3年にわたり行なわれた職業人インタビューは，これから職業を選ぶ生徒はもちろん，すでに職業をもっている大人にも十分に読みごたえのあるものです。

◀「親や親戚以外の働く人を見」て，「仕事や職業について知りたい」。すでに職業に就いている審査員たちも同じ気持ちで，このリポートを読んだことが授賞につながったと思います。

▲取材対象となった歯医者さんの熱いメッセージが伝わってきます。

企画力で受賞！

▶「コミュニケーションする人々」「命を守り・育てる人々」といった分かりやすいジャンル分けが光る職業調査レポート。

学校のホームページ募集　あなたの学校のホームページを紹介します

本誌「マイコンBASICマガジン」では全国の小学校・中学校・高等学校の児童や生徒が関わっているホームページを募集しています。“ぼくの学校も紹介して！”という方は下の応募方法をよく読んで応募してください。また，「取材してほしい」などの希望も受け付けます。本誌掲載の他，ベーマガホームページ上の学校ページリンク集へのリンクなどの特典があります。

Eメールの場合は説明文を添付し，件名に「僕たち私たちのホームページへようこそ！」と記入して，投稿してきてください。

webmas@basicmagazine.net

◆説明文には

①住所，氏名，年齢
②連絡先電話番号，メールアドレス
③学校名（所在地）
④ホームページのURL，内容，PR文
⑤使用パソコン，ツール，ソフト
⑥制作にあたり苦労した点やアピールしたい手法
⑦生徒のみなさんの集合写真（デジタル，アナログどちらでも可）
⑧このコーナーへの要望，感想，質問などを記入してください。

郵送の場合は

〒141-8715　東京都品川区東五反田1-11-15
電波新聞社　マイコンBASICマガジン編集部
「僕たち私たちのホームページへようこそ！」係に説明文を同封して応募してください。

宮城県中田町立上沼中央小学校

http://www2.ocn.ne.jp/~uwanuma/

2002年「第7回わこうどホームページコンテスト」最優秀賞受賞を筆頭に，3年弱の間に10の受賞歴を誇るコンテスト常連サイト。受賞の決め手もさまざまで，ページデザインのよさや視聴覚教材作成，調査レポート，ライブカメラなどなど，本当にどこを切っても多彩でみごとです。

卓抜デザインで受賞！

▶数々の賞に輝き，他校との交流も生まれている自作教材のページは身近な野菜がテーマです。

▼テーブル(表)機能を最大限に生かした充実トップページ。配置や見せ方がみごとでコンテスト受賞歴も納得の高レベルです。

◀実力派サイトはデジカメに頼らない！
生徒の描いた似顔絵作品で先生方を紹介します。

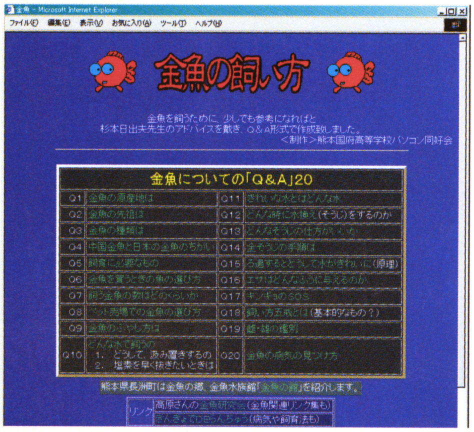
熊本国府高等学校

http://www.kumamotokokufu-h.ed.jp/

2002年「第5回ICFホームページコンテスト」最優秀賞受賞。「熊本の隠れた情報を掘り起こし，全国に紹介しよう！」というテーマで発展し続けている開設8年目の老舗サイトです。地理，歴史，環境などの基本はもちろん，地元産業と密着した超人気コンテンツ・金魚の飼育Q&Aまで，地元をテーマにとことん広がっています。

地元への貢献度で受賞！

▶授業内容に関しては総合学習や社会科などの研究発表が圧倒的に多いスクール・ホームページ界ですが，数学でしかも文章主体のエッセイ仕立てはたいへん珍しいです。



▲大きな画像のないトップページはナローバンドユーザーにも見やすく，機能的にもシンプルで分かりやすいユニバーサルデザインです。

▼地元熊本県の名産でもある金魚をテーマにした「金魚の飼い方Q&A」コーナーは反響が大きかったとか。地元の金魚博物館や個人サイトなどとのリンクもあります。

ICOM
ネットワークは無線に進化
無線LANはアイコム
モデルハウス無線LAN伝送実験を公開!
住宅メーカーモデルハウスでの無線LAN伝送実験を
アイコムホームページ http://www.icom.co.jp/ にて公開中!
WAVEMASTER
広告掲載全機種 Windows® XP対応
80Mbps※超高速ルーティングで、
ADSL/CATVそして光ファイバーまで対応。
通信速度が選べる、
カード交換フレキシブル仕様。
新製品
54Mbps/11Mbps
(IEEE802.11a準拠) (IEEE802.11b準拠)
無線LANカード装着時
54Mbpsまたは11Mbps無線LANの切り替えを実現。
無線端末の規格に合わせて通信速度の設定が可能。
SL-5000
54Mbps (IEEE802.11a準拠)
無線LANカード装着時
光ファイバーの速さを存分に活かせる54Mbpsで、
超高速の無線ネットワークを構築する。
SL-50
11Mbps (IEEE802.11b準拠)
無線LANカード装着時
今は既存のネットワークに合わせて11Mbpsで。
でも将来はカードの付け替えで54Mbpsにできる。
SL-12
UPnP 対応
アイコムホームページから
無償ファームウェアの
ダウンロードが可能です。
Bフレッツ
フレッツ・ADSL
YAHOO! BB
対応
強力セキュリティ
AES 搭載
AESまたはOCB AES対応
無線LANカード
装着時のみ有効です。
※当社テストパターン計測値
業界初! 54Mbps/11Mbps
ワイヤレス通信を実現する無線LANカード。
新製品
すでにお持ちのSR-21BBにも
装着・利用できます!
アイコムホームページより更新ファームウエア*を無償でダウンロードしていただくことができます。
*商品発売時期にアップロードを予定。
強力セキュリティ AES 搭載
OCB AES128bit
54Mbps 11Mbps Dual Band
無線LANカード
SL-5000 オープン価格
●54Mbpsまたは11Mbps無線通信を接続先のアクセスポイントに合わせて自動選択(固定も可能)。
●強力暗号化セキュリティOCB AES(128bit)、WEP(64/128/152bit)を搭載。
●PCカードスロットに簡単装着(CardBus Type II 対応)。
Wireless Broadband Router
SR-21BB
ワイヤレスブロードバンドルータ
SR-21BB 〈本体のみ〉 オープン価格
54Mbps または11Mbps 無線LAN※
光ファイバー ADSL CATV 対応
AES/WEP 暗号セキュリティ※
プリントサーバ機能
4ポートスイッチングハブ
簡単設定ソフト
土日祝日もサポート コールセンター設置
※SL-5000装着時はOCB AES(128bit)、WEP(64/128/152bit)に対応。SL-50、SL-12装着時はAES(128/256bit)、WEP(64/128bit)に対応。
お得なセット品も大好評!
ワイヤレスブロードバンドルータ
SR21Sa+b
54Mbps 11Mbps
新製品
オープン価格
SR-21BB
+
本体装着用
54Mbps/11Mbps
無線LANカード
(端末PC用無線LANカードは別売です。)
●無線LAN速度は設定による切り替えとなります。
ワイヤレスブロードバンドルータセット
SR21S52G 54Mbps
特別セット価格 ¥65,800
SR-21BB
(54Mbps
無線LANカード装備)
端末PC用
54Mbps無線LANカード
ワイヤレスブロードバンドルータセット
SR21S12H 11Mbps
特別セット価格 ¥35,800
SR-21BB
(11Mbps
無線LANカード装備)
端末PC用
11Mbps無線LANカード
ブロードバンドルータ/ルータタイプモデム/
ISDNルータをご利用の方
ワイヤレスアクセスポイントセット 標準価格
AP12SL-E ¥34,800
AP-12
端末PC用
11Mbps無線LANカード
ワイヤレスアクセスポイント
AP-12 〈本体のみ〉 標準価格 ¥25,800
SR-21BBがパソコン誌主催の各賞を受賞しました。
(左)SR-21BBとSL-50のセット商品SR21S52GはソフトバンクパブリッシングDOS/Vmagazine2002年9/1号「Technical Test Labs」にてObserver's Choiceを受賞いたしました。
(右)SR-21BBはインプレス発行DOS/V POWER REPORT2002年9月号において「DOS/V POWER REPORT RECOMMENDED」(パワレポ推奨マーク)をつけられた商品です。
土日祝日もサポート
SR-21BB・AP-12 接続サポートコールセンター
☎06-6795-1770
営業時間 (年末・年始、お盆休みを除く)
平日・土日祝日 9~12時/13~17時
●その他の商品に関するお問い合わせは ☎06-6792-4949へ (平日9~12時/13~17時)
ワイヤレスネットワークシーンをリードする、
アイコムの無線LAN。
強力セキュリティ AES 搭載 128/256bit
無線LANカード
SL-50
標準価格
¥16,800
54Mbps
無線LANカード
SL-12 11Mbps
標準価格 ¥8,900
無線LANカード
SL-120
標準価格 ¥10,800
SL-12
ワイヤレス
LANユニット
SU-12
標準価格 ¥12,800
11Mbps
●SL-12、SU-12は、NTT西日本が提供するフレッツ・スポット(標準プラン)に対応しております。
無線LANカード用PCIバスアダプター 新製品
SA-10PCI SL-5000、SL-50、SL-12、SL-120に対応。
標準価格 ¥7,500
54Mbps、11Mbps
同時通信を実現する、
ルータ機能付き
無線LANアクセスポイント
54Mbps 11Mbps 新製品
デュアルバンド対応
ワイヤレスアクセスポイント
AP-5000 オープン価格
強力セキュリティ AES 搭載
●オフィス、工場、大規模ネットワーク用無線LAN製品も豊富にご用意しております。詳しくは、アイコムホームページでご覧ください。
アイコムのTFT液晶モニター
i-Vista™
15インチ
17インチ
19インチ
ステレオスピーカー内蔵
インテリア性に優れた
スタイリッシュ&スリム。高品質仕様。
15インチTFTカラー液晶モニター
LD-T15GS オープン価格
SXGA対応、
1677万色表示、スイーベル搭載。
17インチTFTカラー液晶モニター
LD-T17E オープン価格
DVI端子搭載、500:1の
高コントラスト、水平・垂直160°広視野角。
19インチTFTカラー液晶モニター
LD-T19 オープン価格
取扱店:アイコム情報機器(株)TEL.06-6793-1101 HP http://www.icom-jk.co.jp/、大手SIer、ソフトアイランド各店、全国有名家電、パソコン・アマチュア無線ショップほか
アイコム株式会社 (東証1部/大証1部上場) 本 社 〒547-0003 大阪市平野区加美南1丁目1-32 ●アイコム製品に関する詳しい情報は、アイコムホームページへ。http://www.icom.co.jp/
北海道営業所 TEL 011-820-3888 仙台営業所 TEL 022-298-6211 東京営業所 TEL 03-5600-0331 名古屋営業所 TEL052-832-2525 大阪営業所 TEL 06-6793-0331 広島営業所 TEL 082-501-4321 四国営業所 TEL087-835-3723 九州営業所 TEL092-541-0211
●表記の価格は税別表示です。●カタログをご希望の方は、製品名、住所、氏名、年令及びご覧になった雑誌名、月号をご記入の上、〒547-0004 大阪市平野区加美鞍作1丁目6-19 アイコム(株)カタログ係まで。●商品の外観及び仕様は予告なく変更する場合があります。●写真は実物と細部が異なる場合があります。●TFTカラー液晶モニターは、画面の一部に点灯しない画素や常時点灯する画素を含む場合がありますが、故障ではありませんので、あらかじめご了承ください。●画面はハメコミ合成です。●アイコム株式会社、アイコム、Icom Inc.、ICOMロゴ、WAVEMASTER、アイドットコム、i-Vistaはアイコム株式会社の登録商標または商標です。●Windowsは米国マイクロソフト社の登録商標です。●その他の社名、製品名及びサービス名は、各社の商標または登録商標です。
インターネットで無線LAN・液晶モニターが購入できるショッピングサイト、アイドットコム®。
http://i.com.icom.co.jp/

記録型パソコン用 DVD技術を知る！

第1回 記録型DVDドライブの選び方

遠藤　正奥

記録型DVDドライブは1998年にパナソニックから記録容量4.7GBのDVD-RAMドライブが発売されていましたが，パソコンのデータストレージとしての位置づけでした。一般のDVDプレーヤで再生できるディスクが焼ける単体のドライブは2001年6月にエスティートレード，ソニー，ロジテックから発売されました。いずれもパイオニア製のドライブを使用しておりDVD-R/RWに対応していました。7月にはパナソニックからDVD-RAM/Rドライブ，9月にはリコーからDVD+RWが発売されました。

2002年にはさらにDVDプレーヤとの再生互換性の向上，多種類の記録型DVDに対応できるドライブなど使い勝手が向上したモデルが各社から発売になりました。付属ソフトも充実し，価格も初期の1/2から1/3なり，オリジナルDVD-Videoが作れる環境が整ってきました。

この連載では数回にわたり，目的にあった記録型DVDドライブの選び方，ビデオ素材の取り込み，編集，書き込みなど，DVDプレーヤで再生できるオリジナルDVD-Videoを制作する具体的な手順を説明していきます。

ビデオディスクの変遷

1970年代から映像をディスクに記録するさまざまな方式が開発され，1972年に登場した信号の読み取りにレーザー光を使用した光学ビデオディスクが，1980年代に主流となり普及していきました。レーザー光により信号を検出するのは現在のCD,DVDと同じですが，記録方式がアナログでディスクのサイズはアナログLPレコードと同じ直径30cmでした。

その後，音声記録は1982年に登場したCDと同じ2chデジタル信号対応（1984年），5.1chのAC-3デジタル信号対応（1994年）と改良されました。本格的に普及したLD（レーザーディスク）の再生時間は片面60分でしたので，映画タイトルは両面再生でも1枚に収まらず3面2枚組が多くありました。

一方，デジタル信号処理技術，デバイスの発達により映像をデジタル化して記録することが可能になってきました。1995年末にDVDフォーラム（東芝，日立製作所，松下電器産業，三菱電機，日本ビクター，パイオニア，フィリップス，ソニー，タイムワーナーの9社）によりDVD規格がまとめられました。当初，映画記録を主な目的に開発されていたため「Digital Video Disc」と言われていましたが，大容量の記録メディアとしてビデオ信号やオーディオ信号，パソコンのデータなどのデジタルデータの記録媒体として多用途あるいは多彩な（Versatile）ディスクメディアとして使おうということで「Digital Versatile Disc」と呼ぶようになりました。

《第1表》ＤＶＤ規格の主な仕様（片面一層記録の場合）

ディスク直径	120ｍｍ
ディスクの厚さ	1.2ｍｍ（0.6ｍｍ×2）
記憶容量	片面4.7ギガバイト
トラックピッチ	0.74 μｍ
半導体レーザ波長	650／635ｎｍ
レンズ開口数	ＮＡ＝0.6
エラー訂正方式	ＲＳ－ＰＣ方式
変調方式	8－16方式

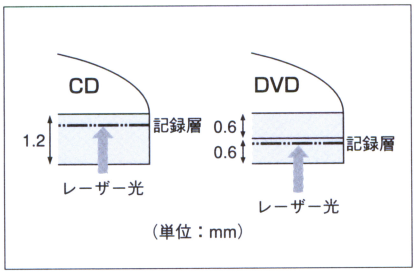
再生専用DVD-ROMの仕様と特徴

前述のようにDVDの規格制定時には，すでにCDが普及していましたので，プレーヤの再生互換とスムーズな普及を考え，外径は12cmと8cm，厚みは1.2mmと，形状はCDと同じにされています。厚み

《第2表》映像再生用の仕様

転送レート	可変方式　平均4.69メガビット（映像＋音声）
画像記録方式	MPEG2画像圧縮
音声記録方式	Dolby　AC－3＊（5.1ch），LPCM（NTSC） MPEG2　AUDIO，LPCM（PAL／SECAM） （最大８か国語の音声と最大32か国語の字幕を記録可能）
収録時間	133分 （転送レート平均毎秒4.69メガビット（映像＋音声）時。 DolbyTMCA－3（5.1ch）3言語4サブタイトルを含む。）
ファイル管理構造	Micro　UDF　および　ISO－9660

＊“Dolby”は，Dolby Laboratory Licensing Corporation の商標です。



《第1図》CDとDVDの断面構造比較図

は同じですが，CDが単版であるのに対して，DVDでは記録容量を増やすために0.6mmのディスクを2枚貼り合わせています。

この構造を生かし長時間再生を可能にするためそれぞれの層に記録するなど，DVDには片面単層，片面2層，両面単層，両面2層の4種類のメディアが規定されています。通常，私たちが手にすることが多い映画ソフトのDVD-Videoは片面単層か片面2層になっています。片面2層で約4時間の記録が可能ですので，ほとんどの映画ソフトがディスク1枚の片面ですんでしまいます。

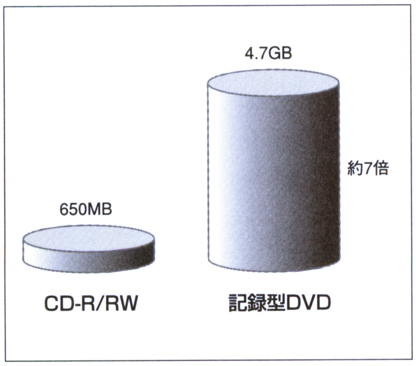
・外形12cmDVDの記録容量

片面単層	4.7GB
片面2層	8.5GB
両面単層	9.4GB
両面2層	17GB

記録型DVD

4.7GBの記録型DVD規格の制定および規格の普及·促進を図っている組織はDVDフォーラムと，DVD+RWアライアンス（リコー，ソニー，ヒューレットパッカード，フィリップス，三菱化学，ヤマハの6社が中心）の二つがあります。

DVDフォーラムのリライタブルにDVD-RW，DVD-RAMがあり，ライトワンスにDVD-Rがあります。DVD+RWアライアンスにはリライタブルのDVD+RWとライトワンスのDVD+Rがあります。この5種類の中でDVD-RAMは当初からパソコンの外部記憶装置としてデータの保存に使うことを主な用途として開発されてきています。動画を保存し，DVDプレーヤで使えるのは再生専用DVD-ROMと再生互換性を求めて開発されたリライタブルのDVD-RW，DVD+RWとライトワンスのDVD-R，DVD+Rになります。



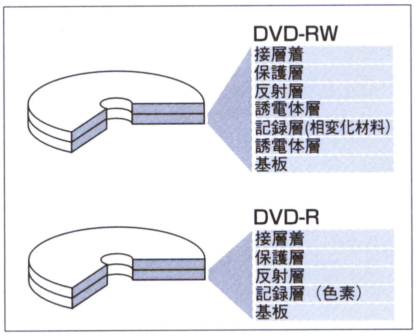
《第2図》CDとDVDの容量比較

ライトワンスDVDディスク

DVD-R，DVD+Rは一度しか書き込めないライトワンスメディアです。記録方式はCD-Rと同じように記録層の色素をレーザー光で焼いて変化させています。反射率もDVD-ROMとほぼ同じ高い再生互換性が得られています。先発のDVD-Rは追記ができない，小容量データでも書き込みに時間がかかるなど当初は使いづらい面がありましたが，書き込みソフトの対応で改善されており，2002年発売のモデルでは，どちらもほぼ同じように使えるようになりました。ライトワンスは消したくない動画やデータの保存，配布用DVD制作に適しています。

リライタブルDVDディスク

DVD-RW，DVD+RWは書き換え可能なリライタブルメディアです。CD-RWと同じ相変化型の光ディスクで，記録層の結晶/非結晶で情報を記録しています。約1000回の書き換えが可能です。反射率が低いのとプレーヤ側での対応も必要なことから，古いモデルのDVDプレーヤ



《第3図》DVD-RWとDVD-Rのディスク構造イメージ図

では再生できないことがあります。互換性についてはDVD-RWはパイオニアのホームページに，DVD+RWについてはリコーのホームページに掲載されていますので参考にしてください。映像データの保存，日々のバックアップなどの一時的なデータ保存や，ライトワンスDVDに書き込む前のテストなど，書き換え型のメリットを活かした使い方ができます。メディアの価格が下がってきたとはいえDVDのライトワンスメディアはCD-Rに比べるとまだまだ高価ですから，失敗しないためにもリライタブルメディアを活用して下さい。

従来は，DVD-Videoとして書き込んだディスクの内容の変更は，すべて消去して最初から書き直さなければなりませんでしたが，最近のモデルでは書き込みソフトの対応で，ディスク上で追記や編集，部分削除などを行っても通常のDVDプレーヤとの再生互換性を維持できるようになってきました。

DVD-RAMは書き換え回数が約10万回もできるリライタブルメディアで，コンピュータのデータ保存には最も適しています。WindowsXPではドライバなしでも認識され，フロッピーやMO感覚で使えデータの保存には最も適したメディアです。

記録型DVDドライブの種類

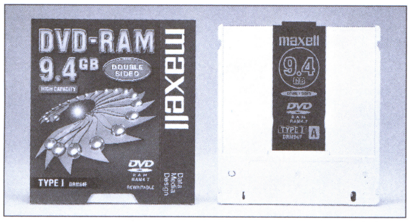
ユーザー側からの理想を言えば，12cmの光ディスクなら目的に応じて，どのディスクでも使えるというのが望ましいところです。2002年の秋に，この理想に一歩近づくことができるようになりました。ソニーからDVD-RW/RとDVD+RW/Rに対応した「DRU-500A」が，また松下電器産業からはDVD-RW/RとDVD-RAMに対応した「LF-D521JD」が発売になりました。対応するメディアが多いと安心感はありますが価格も高くなります。

DVDプレーヤで再生するDVD-Video作成には，DVD-RかDVD+Rが必須です。また，時間の長い映像データを頻繁に扱うとなると書き込み速度も考慮しなければなりません。**第3表**にあげたのは最新モデルですから，どのモデルを選んでも失敗はないと思います。これ以前のモデ

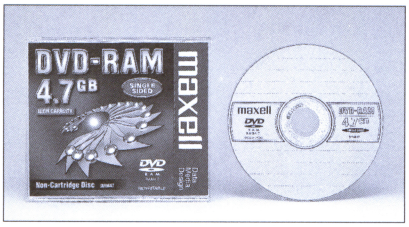
《第3表》代表的モデルが対応する書き込みメディアと書き込み速度

	パイオニア DVR-A05-J	リコー MP5125A	松下電器産業 LF-D521JD	ソニー DRU-500A
DVD-R	4倍速	×	2倍速	4倍速
DVD-RW	2倍速	×	1倍速	2倍速
DVD+R	×	2.4倍速	×	2.4倍速
DVD+RW	×	2.4倍速	×	2.4倍速
DVD-RAM	×	×	2倍速	×
CD-R	16倍速	12倍速	12倍速	24倍速
CD-RW	8倍速	10倍速	8倍速	10倍速

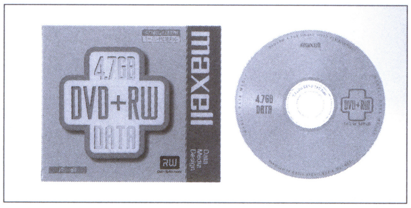
データ記録用として開発された DVD-RAM



▲DVD-RAMカートリッジ入りタイプ（maxell）両面9.4GB／両面5.2GB／片面4.7GB／片面2.6GBがある



▲DVD-RAMカートリッジなしタイプ（maxell）片面4.7GB



▲書き換え可能なリライタブルメディアDVD+RW，DVD-RW（maxell）

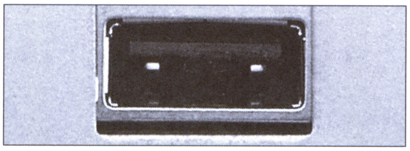
ルでは追記ができない，フォーマットに時間がかかるなど使い勝手の悪い物もあります。

記録型DVDドライブの接続方法

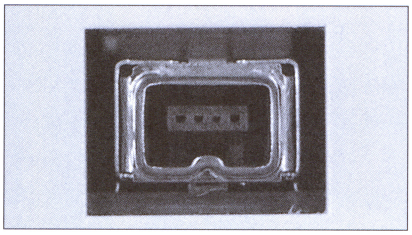
パソコンに記録型DVDドライブを増設しようとするとき，接続方法には以下の3とおりの方法があります。

内蔵型　ATAPI接続

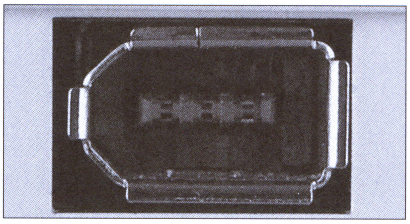
パソコンの端子群をチェック！



▲USBコネクタ



▲IEEE1394（4ピン）



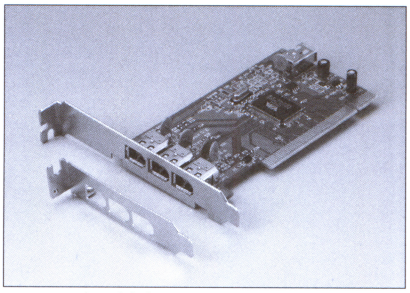
▲IEEE1394（6ピン）

外付け型　USB2.0接続

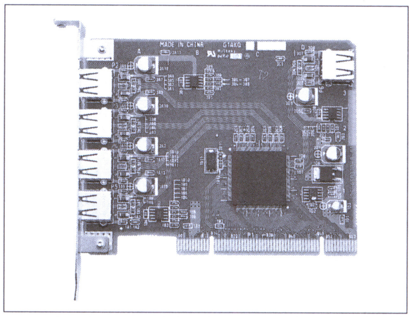
外付け型　IEEE1394接続

IEEE1394はソニーではiLINKと呼んで，デジタルビデオカメラなどに使われています。どの接続方法を選ぶかは，いろいろな条件で変わってきます。パソコンがタワーなのかノートなのかという形態の問題があります。

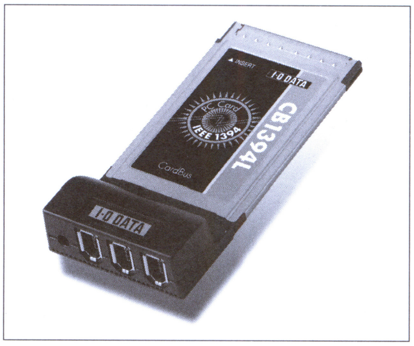
タワーなら内部の増設スペース，接続端子の空き，ケースを開けて取り付ける技量の有無などによって，内蔵にするか，外付けにするかを決めます。外付けの場合はパソコンにIEEE1394またはUSB2.0の端子が必要になります。端子がない場合はインターフェースボードを取り付けなければなりません。このときもパソコンのケースを開けての作業になりますが，ドライブの取り付けよりは簡単です。もちろんPCIスロットの空きがなければ増設はできません。



▲PCIバス対応IEEE1394インターフェースボード（メルコ「IFC-ILP4」価格3,300円）



▲USB2.0&USB1.1 PCIバス用インターフェースボード（メルコ「IFC-USB2P5」価格3,300円）



▲CardBusスロット用IEEE 1394インターフェースPCカード（アイ・オー・データ機器「CB1394L」価格5,500円）

ノートの場合は外付けになります。最近ではIEEE1394とUSB2.0の端子をもったモデルも多くなってきましたが，ない場合にはCardBusスロット用のPC カードタイプのインターフェースが必要になります。

また，複数のパソコンに繋ぎ替えて使いたいときも外付けになります。

外付け型のインターフェースを新規に用意するのであればIEEE1394がお勧めです。最近の周辺機器はUSB対応が多くなっていますが，デジタルビデオカメラからのキャプチャーが，この端子があれば1本のケーブルで，しかも高品位にできるという大きな特徴があります。

写真撮影協力＝九十九電機

ソニー

DRU-500A（内蔵型ATAPI）
DRX-500UL（外付け型iLINK/USB2.0）

※IEEE1394をソニーでは iLINKと呼ぶ

DRU-500A

DRX-500UL

DVD+RW，DVD-RW，DVD+R，DVD-R，CD-RW，CD-Rの6種類の書き込みフォーマットに対応しています。DVDフォーラムとDVD+RWアライアンスの両方の規格に1台のドライブで対応したのは初めて。DVDプレーヤで再生互換のある記録型DVDメディアすべての書き込みができ，しかも各メディアへの書き込み速度はDVD+RW/+R 2.4倍速，DVD-R 4倍速，DVD-RW 2倍速，CD-R 24倍速，CD-RW 10倍速と，すべてのフォーマットで高速書き込みを実現しています。DVD-Videoの制作はもとより音楽CDの制作からデータの保存まで，これ1台で文句なしに使えます。ただ高機能な分だけ価格も高めです。

付属ソフト

● MyDVD
ドラッグ&ドロップの簡単操作でDVDが手軽にできるオーサリングソフト。メニュー画面の作成も簡単に行えます。

●ShowBiz
DVカメラなどから取り込んだ映像に，さまざまな特殊効果やサウンドを加えたり，タイトルを挿入するなど，凝った映像編集ができる本格的な映像編集ソフトです。レンダリングはリアルタイムでおこなえるので待たされることがなく，作業時間を大幅に短縮できます。

● RecordNow
DVDフォーマットにも対応したライティングソフト。データの保存やバックアップディスクの作成ができます。

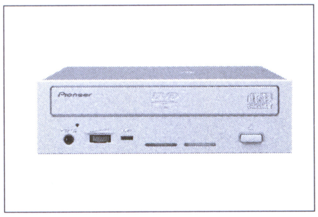
● SimpleBackup
ファイルをそのまま，フォルダ構成もそのままにして指定フォルダにコピーする簡単なバックアップツール。ファイルを圧縮したり，一つにまとめたりしないので，本ソフトがなくても，デスクトップ上で簡単に復元したり，内容を確認したりできます。

● PowerDVD
DVD-VideoやVideoCDの再生ソフト。可変速再生やマルチアングル，マルチ言語，マルチ字幕などに対応しています。

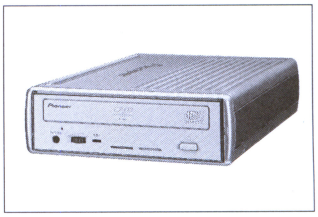
● MUSICMATCHjukebox
各種フォーマットに対応した音楽再生・録音ソフト。音楽データの取り込みやプレイリストの作成，MP3へのエンコードなどができます。

パイオニア

DVR-A05-J（内蔵型ATAPI）
DVR-S502（外付け型 IEEE1394）



DVR-A05-J



DVR-S502

DVD-R書き込み4倍速，DVD-RWも2倍速と前作のDVR-A04JからCD-R/RWも含めて，すべてのメディアで書き込み速度が2倍になっています。DVD-Rへの4倍速書き込み，DVD-RWへの2倍速書き込みは新しい規格で，メディアも新規格の速度に対応したものが必要になります。従来のDVD-Rディスクでは2倍速記録，DVD-RWディスクでは等速記録となります。

QuickGrowという機能によりファイナライズされたDVD-RWメディアを再び追記·編集が可能な状態に戻せるようになり，最初からすべて書き込み直さなくてもすむようになりました。さらに，DVD-Rへの追記のマルチボーダー，1GB以下のファイル書き込みの高速化，DVD-RWのクイックフォーマットなど作業性の向上が図られています。外付け型のDVR-S502も用意されています。

付属ソフト

●DVDit！ SE
本格的オーサリングソフト。簡単なメニューから高度なオリジナルデザインのメニューまで使いこなし甲斐のあるソフト。

●MyDVD
映像のキャプチャーからエンコード，ディスクの書き込みまでをVTRダビング感覚で気軽に行え，簡単にDVD-Videoの作成ができるオーサリングソフト。

●ShowBiz
今回，新たに追加された，ストーリーボードはもちろん，プロ仕様のタイムライン機能も備える本格的な映像編集ソフト。

●インスタントCD＋DVD LE
定評のある本格的なDVD／CD用データ記録ソフト。好評の「クイックフォーマット」をサポートしたDVD-RWパケットライト機能やマルチボーダーをサポートしたデータ記録機能を含む統合型のデータ記録ソフト。

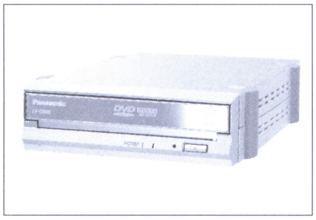
●Drag’nDropCD Plus DVD Edition
シンプルな4ボタンインターフェイスにより，ファイルや映像をドラッグ・アンド・ドロップすることでディスクの作成が行なえる，映像・データ記録ソフト。

●PowerDVD XP
豊富な機能をもつDVD-Video再生ソフト。DVDレコーダのVRモードで録画したディスクも再生可能。

松下電器産業

LF-D521JD（内蔵型ATAPI）
LF-D560JD（外付け型USB2.0）

LF-D521JD



LF-D560JD

DVDフォーラムが規格化した記録型DVDフォーマットのDVD-RAM，DVD-R，DVD-RWのすべて(DVD MULTI規格)と,CD-R・CD-RWの記録・再生に対応したDVD MULTIドライブです。

両面で9.4GBのDVD-RAMをはじめDVD-R，DVD-RW，CD-R，CD-RWと多くの記録型メディアに対応していますので，映像データ，音楽データ，PCデータなど，それぞれの用途に適したディスクの選択ができ，効率的なデータの保存ができます。

本機は大量のデータをスピーディに書き込めるように，今回新たに通常DVD-RAMで行っていたベリファイ機能をOFF（他の記録ディスクと同様）にすることで，通常の2倍の高速記録を実現しています。さらに，バッファアンダーラン防止機能（Smooth Link）搭載で，DVD-R・DVD-RW・CD-R・CD-RWの書き込み時にまれに起こる転送エラーの対策を行なっています。

付属ソフト

●DVD-MovieAlbumSE 3
映像のキャプチャーから記録・編集，DVDオーサリングまでが簡単に行えるソフト。VR（ビデオレコーディング）規格によりDVD-RAMディスクで映像を記録・編集することができます。

●MyDVD 3.5
映像のキャプチャーからエンコード，ディスクの書き込みまでを気軽に行え，簡単にDVD-Videoの作成ができるオーサリングソフト。ドラッグ&ドロップの操作で簡単にメニュー画面の構築ができます。

●B’s Recorder GOLD5 BASIC
オリジナルのデータCD・DVD・オーディオCDの作成など，多彩な機能を備えたライティングソフト。DVD-RAMでは高速記録（ベリファイON/OFF選択可能）に対応しています。

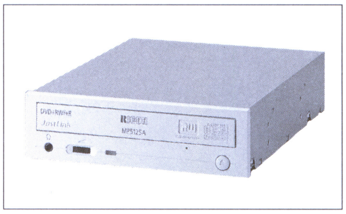
● B’s CLiP5
DVD-RW・CD-RWディスクにファイル単位でデータを記録するためのパケット記録ソフト。

●Win DVD 4
ソフトウェアDVDプレーヤ。高画質・高音質のDVD-Videoタイトルの再生が可能です。

●FileSafe
指定したフォルダを任意のスケジュールで，DVD-RAMに自動でバックアップする簡易バックアップユーティリティ。

リコー

MP5125A（内蔵型ATAPI）



MP5125A

DVDプレーヤ等との，より高い互換性が得られるDVD+Rディスクの書き込みも行えるようになりました。

DVD+RW，DVD+Rともに書き込み速度は2.4倍速で，リライタブルDVDメディアでは一番速度が速く，大量のデータの保存を得意としています。

DVD+RWビデオレコーディングフォーマットに対応したことにより，付属のオーサリングソフトを使うことで，DVDプレーヤと再生互換を保ったままビデオクリップの追加，編集が可能になっています。

永年のCD-R/RWの開発で培われてきた技術が生かされ，追記･上書き時に既存のデータと僅か1ミクロン以下の高精度で，ほぼつなぎ目のない書き込みができるロスレスリンキング技術など，信頼性の高いドライブに仕上がっています。

付属ソフト

●B’s CLiP3.2
CD-R/RWやDVD+RW/+Rへのデータ書き込みを実現するパケットライトソフト。

●B’s Recorder GOLD3.2
さまざまなフォーマットのCD-ROMや音楽CD，さらにDVD-ROMも作成できるマスタリングソフト。

●neoDVD standard v2.6
簡単操作と多彩な機能を実現したDVD-Videoオーサリングソフト。

● Drag’nDropCD Ver.2
書きたいファイルをアイコンにドラッグ&ドロップするだけでCD-ROMから音楽CD，DVD-ROM，DVD-Videoまでを簡単に作成できるライティングソフト。

● WinProducer 2
ビデオクリップのカット編集や並び替え，特殊効果などが可能。AVIファイルだけでなく，MPEG形式のファイルも編集可能なビデオ編集ソフト。

● WinDVD3.2
DVD-Videoプレーヤソフト。

● earjam Internet Music Player
MP3やWMAなどの音楽ファイルを再生したり，CDやHDD，さらにはネットワーク上に書き込むことができるミュージックソフト。

●DVD-Videoの作成に便利な自由に使用できるムービークリップ集。

インフォ・キャッチ Information Catch Up 02

パソコン 液晶パネルから音が出る新スピーカー・システム NEC VALUESTAR FS

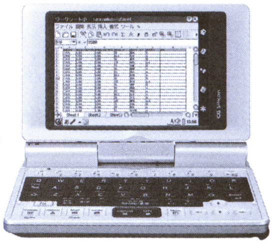
●液晶一体型パソコン
オープン価格
（実売価格：23万円前後）
CPU：Celeron1.7GHz
メモリ：256MB
ハードディスク：約80GB
ディスプレイ：17インチ・ワイド
ワイヤレス・キーボード，赤外線リモコン付属
■問い合わせ先
☎0120-977-121
http://121ware.com/valuestar/

パソコン 組み合わせ自由なWeb販売専用パソコン NEC VALUESTAR Gタイプ A

▲写真は17インチ・ダイヤモンドトロン・モニタをセットにしたVG26N4/8

●ミニタワー型パソコン
価格：8万800円より
（Web販売価格／モニタ別）
CPU：Duron1.1GHzなど
メモリ：256MBより
ハードディスク：60GBより
■販売サイト
http://121ware.com/store/
■問い合わせ先
☎03-5762-0450
http://121ware.com/

PDA 液晶部分が回転・反転，タテ型ヨコ型で使える シャープ ザウルス SL-C700



オープン価格（実売価格：5万円台後半）
OS：Linuxベース
メモリ：フラッシュ64MB，SDRAM32MB
ディスプレイ：3.7インチ透過型液晶
画面解像度：640×480ドット
記録メディア：SDメモリカード，コンパクト・フラッシュ
電源：リチウムイオン充電池（約4時間50分）
■問い合わせ先
http://sl.ezaurus.com/index.html

ノートパソコン 3.5mmストローク・キーボードでデスクトップなみの快適操作 シャープ Mebius PC-FS2-C3M/C3E

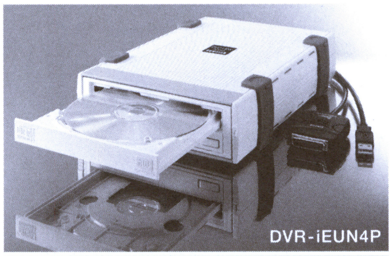
●A4ノート・パソコン
CPU：モバイルAthlon XPプロセッサ 1400+
メモリ：256MB
［PC-FS2-C3M］
オープン価格
（実売価格：16万5,000円前後）
ハードディスク：40GB
ディスプレイ：15インチ
［PC-FS2-C3E］
価格：オープン価格
（実売価格：16万円前後）
ハードディスク：30GB
ディスプレイ：14.1インチ
■問い合わせ先
☎043-299-8021（東日本相談室） ☎06-6794-8021（西日本相談室）
http://www.sharp.co.jp/mebius/index.html

ルータ 通信速度が選べる無線LANブロードバンド・ルータ アイコム SR21Sa+b

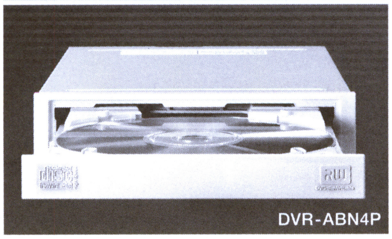
ブロードバンド・ルータSR-21BBと無線LANカードSL-5000のセット
価格：オープン価格（実売価格：3万4,000円前後）
通信速度：11Mbps／54Mbps選択可
対応OS：Windows 98/98SE/Me/2000/XP
■問い合わせ先
☎06-6795-1770
http://www.icom.co.jp/

DVD+Rドライブ 4倍速対応のDVD+Rドライブ アイ・オー・データ機器 DVR-iEUN4P/DVR-ABN4P

［DVR-iEUN4P（外付型）］
価格：3万7,500円
（実売価格：3万円前後）
［DVR-ABN4P（内蔵型）］
価格：2万8,500円
（実売価格：2万5,000円前後）
対応メディア：［書込み］DVD+R（4倍速），DVD+RW（2.4倍速），CD-R（16倍速），CD-RW（10倍速），
［読出し］DVD-ROM（12倍速），DVD+R/RW（5倍速），CD-RW（32倍速），CD-ROM（40倍速）
インターフェース：ATAPI，IEEE 1394，USB1.1／2.0
■問い合わせ先
☎03-4288-1039
http://www.iodata.jp/



DVR-iEUN4P



DVR-ABN4P

ホームPC用家具 パソコンを載せてスイスイ移動できるカート型デスク 岡村製作所 パソランド スキッピィ

［パソランド スキッピィカート］
（ディスプレイ取り付けタイプ／価格：2万9,800円）
［パソランド スキッピィワゴン］
（ディスプレイ置き型タイプ／価格：2万4,000円）
■問い合わせ先
☎0120-81-9060
http://www.okamura.co.jp/

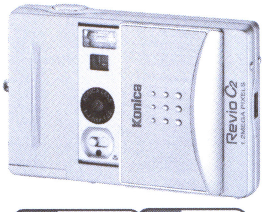
■お正月から卒業式まで。思い出いっぱいデジカメで写そう■

有効画素数 200万画素　ズーム 単焦点

200万画素の超薄型軽量デジカメ

カシオ計算機　EXILIM(エクシリム) EX-S2

オープン価格（実売価格：3万円前後）
記録メディア：SDメモリカード，マルチメディア・カード
サイズ：幅88mm×高55mm×奥行11.3mm
重量：88g
■問い合わせ先
☎ 0570-088902
http://www.exilim.jp/



有効画素数 122万画素　ズーム デジタル2倍

カード・サイズのコンパクトなデジカメ

コニカ　Reivio(レビオ) C2

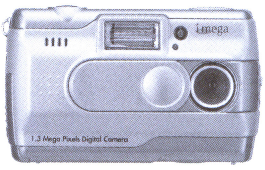
オープン価格（実売価格：2万円前後）
記録メディア：14MB内蔵フラッシュ・メモリ
サイズ：幅85.8mm×高55.8mm×奥行13.5mm
重量：70g
■問い合わせ先
☎ 042-587-1044
http://www.konica.co.jp/personal/index.html

有効画素数 201万画素　ズーム デジタル4倍

フェースパッドで着替えるオシャレなデジカメ

東芝　sora(ソラ) T15

オープン価格（実売価格：3万円前後）
記録メディア：SDメモリカード
サイズ：幅85.5mm×高72mm×奥行27.9mm
重量：120g
■問い合わせ先
ナビダイヤル 0570-05-7000
http://www.toshiba.co.jp/mobileav/camera/index_j.htm



有効画素数 131万画素　ズーム デジタル2倍

手のひらサイズのお手頃デジカメ

日立リビングサプライ　i.mega(アイ.メガ) HDC-1

オープン価格（実売価格：1万3,000円前後）
記録メディア：8MB内蔵フラッシュ・メモリ，SDメモリカード
サイズ：幅93mm×高56mm×奥行38mm
重量：110g
■問い合わせ先
0120-880228
http://www.hitachi-ls.co.jp/shinjyouhou/index.html

有効画素数 320万画素　ズーム 光学3倍 デジタル4倍

フルフラットなのに光学3倍ズーム

ミノルタ　DiMAGE(ディマージ) Xi

オープン価格（実売価格：5万5,000円前後）
記録メディア：SDメモリカード，マルチメディア・カード
サイズ：幅84.5mm×高72mm×奥行20mm
重量：130g
■問い合わせ先
ナビダイヤル 0570-007111
http://www.dimage.minolta.co.jp/xi/index.html

有効画素数 320万画素　ズーム 光学3倍 デジタル3倍

ライカとの共同開発で実現した高画質

松下電器産業　LUMIX(ルミックス)-F1

価格：4万9,800円（実売価格：4万5,000円前後）
記録メディア：SDメモリカード，マルチメディア・カード
サイズ：幅104mm×高50.9mm×奥行32.1mm
重量：160g
■問い合わせ先
0120-878-3650
http://www.panasonic.co.jp/products/dc/

有効画素数 322万画素　ズーム 光学3倍 デジタル4倍

小型・軽量・高性能！

京セラ　Finecam(ファインカム) S3L

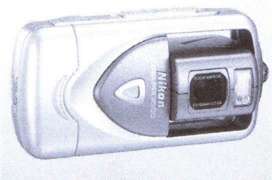
価格：6万2,000円（実売価格：5万5,000円前後）
記録メディア：SDメモリカード，マルチメディア・カード
サイズ：幅92mm×高57.5mm×奥行33mm
重量：165g
■問い合わせ先
☎ 0077-78-0500
http://www.kyocera.co.jp/prdct/optical/

有効画素数 400万画素　ズーム 光学3倍 デジタル4倍

動画も高画質なデジカメ

三洋電機　DSC-AZ3

価格：8万7,000円（実売価格：5万5,000円前後）
記録メディア：コンパクトフラッシュ・カード(TYPE I／II)，マイクロドライブ
サイズ：幅99mm×高55mm×奥行32.5mm
重量：210g
■問い合わせ先
0120-398634
http://www.sanyo-dsc.com/

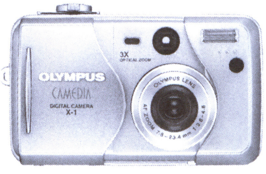


有効画素数 320万画素　ズーム 光学3倍 デジタル4倍

小型・軽量・高性能！

ニコン　COOLPIX(クールピクス) 3500

価格：7万2,000円（実売価格：4万5,000円前後）
記録メディア：コンパクトフラッシュ・カード（TYPE I）
サイズ：幅114mm×高59.5mm×奥行31.5mm
重量：165g
■問い合わせ先
ナビダイヤル 0570-02-8000
http://www.nikon-image.com/jpn/index.htm



有効画素数 430万画素　ズーム 光学3倍 デジタル4倍

動画も高画質なデジカメ

オリンパス光学工業　CAMEDIA(キャメディア) X-1

価格：8万1,000円（実売価格：5万円前後）
記録メディア：xDピクチャーカード
サイズ：幅99mm×高55mm×奥行32.5mm
重量：210g
■問い合わせ先
0120-084215
http://www.olympus.co.jp/

※1月号では，この他にフジフィルム『FinePix A203』，ソニー『DSC-U20』を紹介しています。

デジカメで撮った画像をダイレクト・プリントする ― "DPS" 規格

DPS規格に参画しているのは，キヤノン，富士写真フイルム，ヒューレット・パッカード，オリンパス光学工業，セイコーエプソン，ソニーの6社。

現在，デジカメとプリンタを直接つないでプリントする機器は，各社とも独自の方式のため，組み合わせが限定されている。DPS規格が広まれば，どのデジカメとプリンタでも，簡単な操作でプリントできるようになる。

DPS Version1.0は，インターフェースにUSB，データ転送プロトコルにPTP（Picture Transfer Protocol）を採用する予定。
http://www.camera-direct.info/index_j.html

理科実験ソフト

理科実験のシミュレーション・ソフト

アスキーソリューションズ　クロコダイル実験室：物理・理科編 アカデミック版

中学，高校で活用できる理科教育用の実験シミュレーション・ソフトウェア。「電気・電子」，「運動と力」および「光学」の3分野の実験をパソコン画面の中で行えます。

価格：アカデミック版9,800円，通常パッケージ版19,800円，スクールライセンス版19ライセンスまで9,000円，39ライセンスまで7,000円，40ライセンス以上6,000円
開発元：Crocodile Clips（イギリス）
対応OS：Windows 95/98/Me/2000/NT4.0/XP
ハードディスク空き容量：30MB以上
■問い合わせ先
☎ 03-4524-6001
e-mail：school@asciisolutions.com
http://www.asciisolutions.com/
・Crocodile Clipsのホームページ（英語）
http://www.crocodile-clips.com/

マンガ製作ソフト

タブレットPCに対応したマンガ作画の入門ソフト

セルシス　ComicStudioDebut

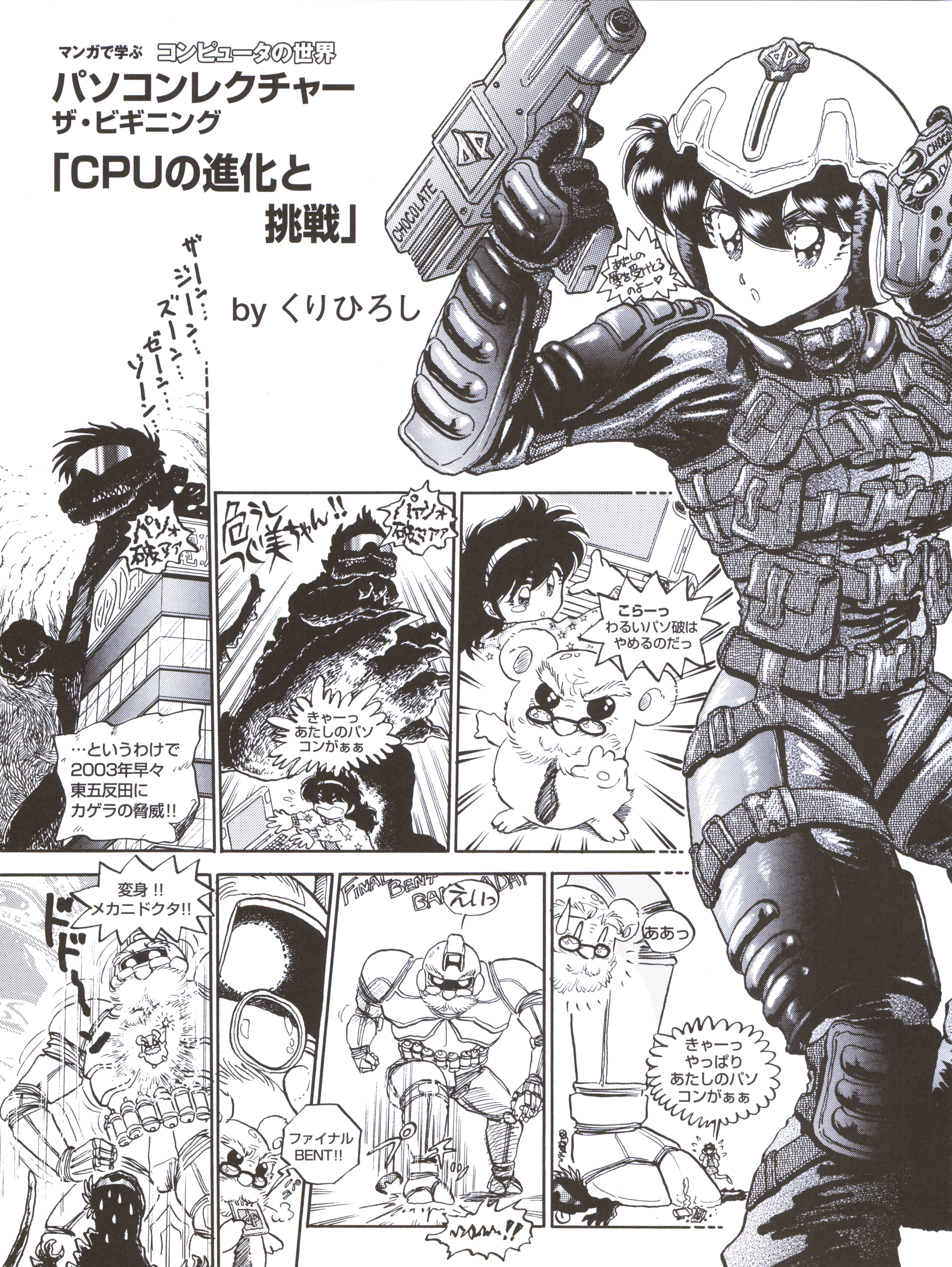
価格：1万1,500円
対応OS：Windows 98SE/Me/2000/XP/XP Tablet PC Edition
ハードディスク空き容量：450MB以上
■問い合わせ先
☎ 03-5304-0855
http://www.comicstudio.net/

マンガで学ぶ　コンピュータの世界

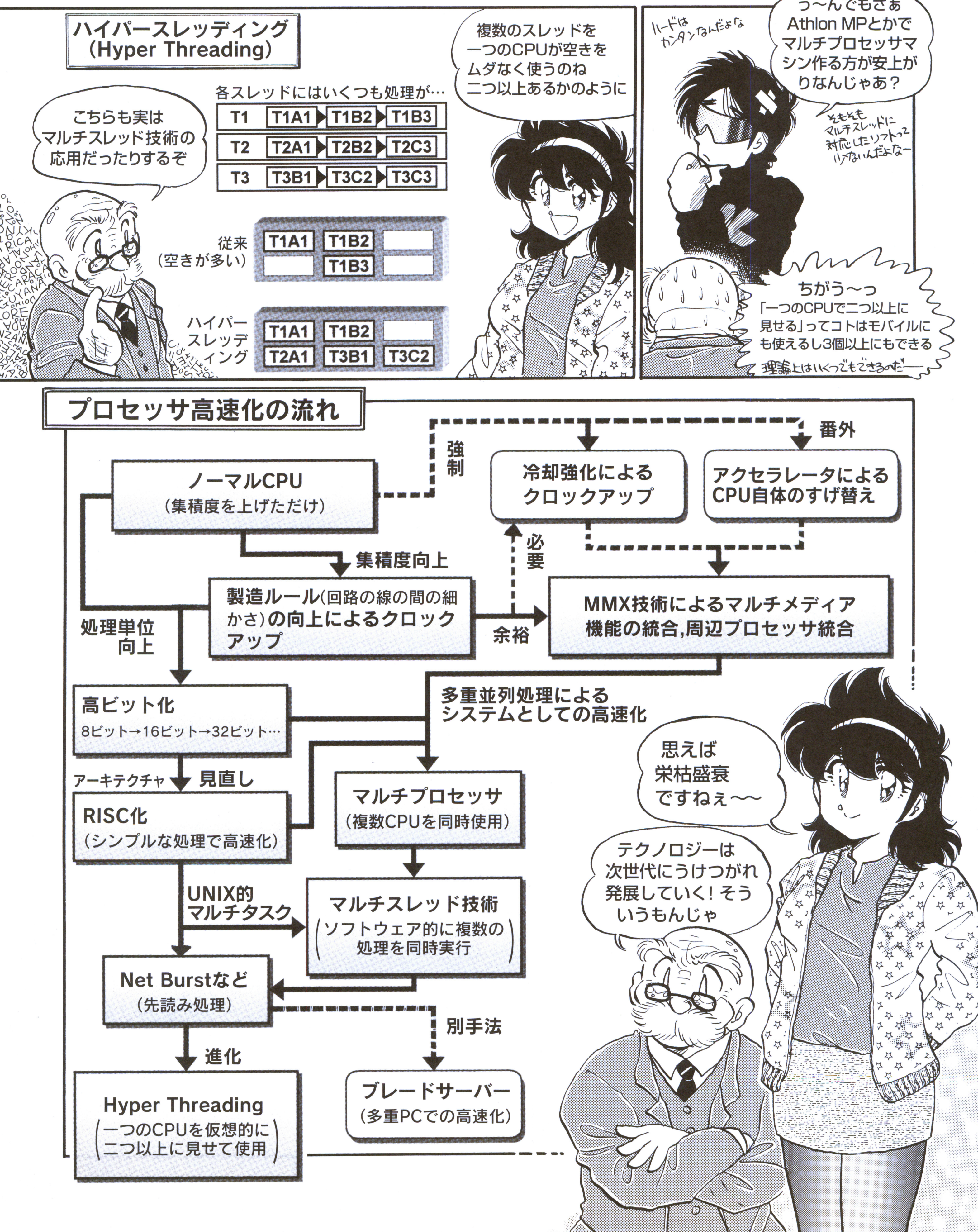
パソコンレクチャー ザ・ビギニング

「CPUの進化と挑戦」

by くりひろし



うわあああんっ
お年玉で買ったスタイリッシュパソコンがぺっちゃんこおぉ～ぉ～
わーっ わかったから泣くのはやめてくれえっ
…編集部のおカネで新しいの買ってくれる？
ぐすん
チラ
ううぅっ まぁ そこは編集長に交渉するわいっ
よおおーし それならやっぱし当然最新技術搭載のハイエンドプロセッサのマシンよねーっ
BABANGA BANG!!
マッタク…
ホント ゲンキンだよなーっ
K
マイクロプロセッサ最新テクノロジー
2002年末！ 久々にわかりやすいマイクロプロセッサの進化が!!
intel
3GHz
2GHz
1GHz
ついに誕生 3ギガヘルツを超えるCPU!!
…とは言っても単純なベンチマークでは2.8GHzに比べて5%程度のスピードアップしかない
ベーマガ独自 CPUパフォーマンス
3.0GHz
6600
2.8GHz
6200
小さい
これはイマイチ!! 3GHz以外のとりえは無いのか!?
実は…
実体はひとつでも見かけ上は二つのCPUに見える,論理上のマルチプロセッサ…
見かけのクロック周波数よりずっと高速に動けるよ。
intel pentium 4
intel pentium 4
ハイパースレッディングテクノロジー採用なのです!!
これまでのCPUの高速化の大きな流れとしては
①クロック周波数を上げる
②CPUを複数並べる
という感じだったのだが…
Pentium 4あたりから採用されはじめた技術は,「高速化されたこと自体を利用してさらに高速化」するものなのだ
初代Pentium 4で使われている「Net Burst」による高速化
A1 A2 A3 B1 B2 B3 C1 C2 C3
通常は順番に処理する
分岐の前を
A1 A2 A3 ? B1 C1
処理中に…
並列して…
C1 C2 C3
分岐を予測して,先に処理してしまう。
（もちろん予測がハズれたらやりなおし，というか高速化されなかったことに）
マルチスレッド（スレッドとは処理単位）という並列処理技術をうまくつかって処理の分かれる先（分岐）
をどんどんやっておくってことよね。確かにこれもある程度CPUが高速化されてないとできないわ



ハイパースレッディング
(Hyper Threading)
こちらも実は
マルチスレッド技術の
応用だったりするぞ
各スレッドにはいくつも処理が…
T1 T1A1 T1B2 T1B3
T2 T2A1 T2B2 T2C3
T3 T3B1 T3C2 T3C3
従来
(空きが多い)
T1A1 T1B2
T1B3
ハイパースレッディング
T1A1 T1B2
T2A1 T3B1 T3C2
複数のスレッドを
一つのCPUが空きを
ムダなく使うのね
二つ以上あるかのように
ハードはカンタンなんだよな
う～んでもさぁ
Athlon MPとかで
マルチプロセッサマシン作る方が安上がりなんじゃあ?
そもそもマルチスレッドに対応したソフトって少ないんだよな～
ちがう～っ
「一つのCPUで二つ以上に見せる」ってコトはモバイルにも使えるし3個以上にもできる
理論上はいくつでもできるのだ～
プロセッサ高速化の流れ
ノーマルCPU
(集積度を上げただけ)
強制
番外
冷却強化によるクロックアップ
アクセラレータによるCPU自体のすげ替え
必要
集積度向上
製造ルール(回路の線の間の細かさ)の向上によるクロックアップ
余裕
MMX技術によるマルチメディア機能の統合,周辺プロセッサ統合
処理単位向上
高ビット化
8ビット→16ビット→32ビット…
多重並列処理による
システムとしての高速化
アーキテクチャ 見直し
RISC化
(シンプルな処理で高速化)
マルチプロセッサ
(複数CPUを同時使用)
UNIX的
マルチタスク
マルチスレッド技術
(ソフトウェア的に複数の処理を同時実行)
Net Burstなど
(先読み処理)
別手法
進化
Hyper Threading
(一つのCPUを仮想的に二つ以上に見せて使用)
ブレードサーバー
(多重PCでの高速化)
思えば
栄枯盛衰
ですねぇ～～
テクノロジーは
次世代にうけつがれ
発展していく! そう
いうもんじゃ

さぁーて2003年以降考えられる路線は何かのう
AMDもトランスメタも,いろいろと新しいテクノロジーを出してきてますもんねぇ
これらは2002年に発表された次世代CPU向け新技術だが…
・トランスメタ新型crusoe
低消費電力かつ内部256ビット
・ハードウェアでのセキュリティ強化
intel 「La Grande」
・本格的64ビット化へ
intel IA-64,AMD×86-64
・低消費電力と高い性能を両立
intel「Banias」
2003年にかけてどれだけ実用化されるかは楽しみな所じゃ
つぐ美ちゃん的にはモバイル大好きで…今年は無線LANがアタリマエになるのよね
ケータイ機能を持つPDAとか期待できるからモバイル向けプロセッサの性能アップが楽しみね!!
発展としては将来的には家庭でもブリッドコンピューティングとか分散CPUによる高速化ができるかも
すでにIBMはこの技術の応用で人間の脳を上まわる処理スピードをたくさんのCPUを並列動作させて実現する予定なのだ
おぉ! そういえば…
すっかり忘れておったが…! つぐ美くん!!
今年はお年玉もあげたんだし…チョコレートはどうしたかのう?
年も年だし なかなか他の人からは もらえんからの～
あー… 忘れてませんよ もちろん
用意はしてたけど…
実はさっきペチャンコにされたPCが…なんと!
バレンタインチョコだったのですぅ!!
だから～ コレ ギャうたしてっ
バ
新しい パソコン なんていらないのでは…
うぅぅっ 何なんだその形はぁ～!?
このへんは チロルチョコ。
TO BE CONTINUED…

◎20点確保はまちがいなし!!

集中連載その2　東幸太

数学①②「BASIC言語」の予想問題と解法

「平成15年度大学入試センター試験」対策直前マスター

いよいよ，センター試験まで10日余り，追い込みの時期である。この時期には新しい内容に手を出すより，すでに学習した範囲の理解度を高め，学習した範囲であればパーフェクトな解答ができるようにするのが得策である。とくにBASIC関連については出題範囲が限られているため，修得内容を充実させるのが効果的である。これまで学習した内容をもう一度復習し，自分のものにしておいて欲しい。

抑えておくべきポイント

センター試験のBASIC問題では，以下の点を抑えておくことが重要である。(1)から(3)については，試験問題に組み込まれることが非常に多い。(4)から(6)は出題可能性は若干低いものの，抑えていないと対処が難しいので要注意である。

(1) FOR−NEXTのループ
(2) IF−THENの条件判断
(3) INT関数の動作とINT関数利用による剰余計算・約数判断
(4) サブルーチンの動作
(5) 配列変数の利用
(6) フローチャートとプログラムの対応関係

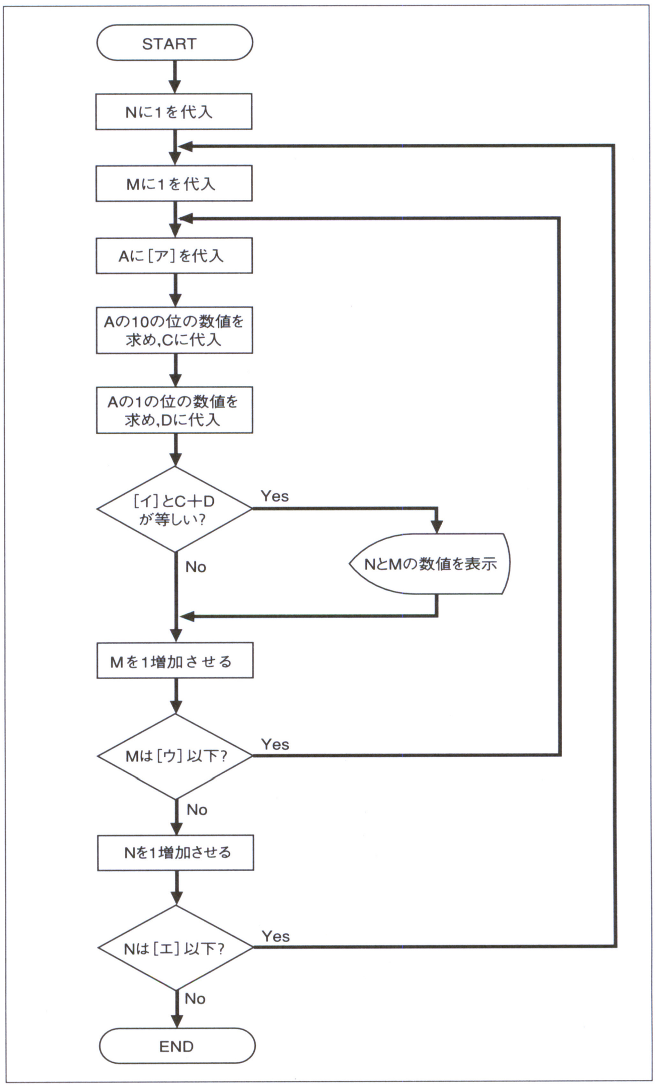
予想問題3

任意の1桁の自然数nとmを掛け合わせた結果のaについて，10の位の数値と1の位の数値を加算した値bを計算し，n＋mとbが等しいn，mの組み合わせを，すべて表示するプログラムを作成したい。

[1] 上記の動作を行うプログラムのフローチャートを作成した**(第1図)**。アからエまでに相応しいものを以下から選び，フローチャートを完成させよ。

アとイに関する選択肢

(1) C＋D　**(2)** C−D　**(3)** C×D　**(4)** C÷D　**(5)** N＋M



《第1図》動作を行うプログラムのフローチャート

(6) N－M　(7) N×M　(8) N÷M

ウとエに関する選択肢

(1) 0　(2) 1　(3) 9　(4) 10　(5) A　(6) C　(7) D

［2］フローチャートを元に以下のようなプログラムを作成した。オからキまでに相応しいものをそれぞれの選択肢の中から選び，プログラムを完成させよ。

```
10 FOR N=1 TO 9
20  FOR M=1 TO 9
30   A= オ
40   C=INT(A/10)
50   D=A- カ
60   IF N+M キ C+D THEN GOTO 80
70   PRINT "n=";N;":m=";M
80  NEXT
90 NEXT
100 END
```

オに関する選択肢
(1) N＊M　(2) N＋M　(3) A＊N　(4) A＊M　(5) A＋1
(6) N＋1　(7) M＋1

カに関する選択肢
(1) N＊10　(2) M＊10　(3) A＊10　(4) C＊10　(5) N/10
(6) M/10　(7) A/10　(8) C/10

キに関する選択肢
(1) >　(2) <　(3) <>　(4) ＝　(5) >=　(6) <=

［3］このプログラムを実行した場合，3行目に表示されるのは，n＝ク：m＝ケである。また，実行終了までにコサ行が表示される。

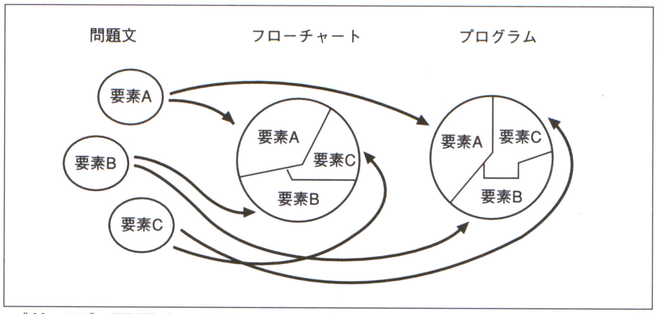
解法3

センター試験で問題となるのはBASICプログラムそのものだけでなく，プログラムと対応するフローチャート（流れ図）も対象となる。この問題はフローチャートを作成させた後，プログラムに関する設問に移行していくパターンとなっている。

設問1では，フローチャートを完成させることが要求されている。フローチャートはプログラムの動作を言葉や図によって示したもので，基本的にプログラムと直接対応する関係になる。

フローチャートの内容は，問題の冒頭で示されている処理と一致するはずである。よって，問題が要求している処理内容を予想し，フローチャートの中ですでに示されている部分との対応をとりながら空欄を埋めていくという作業になる。

このような場合，まずは問題となっている処理に含まれている細かい部分的な要素をリストアップするのが有効である。この問題の場合，



《第2図》問題文で示された要素は,何らかの形でフローチャートやプログラムに含ませる必要がある

(a) 1桁の自然数nとm
(b) nとmを掛け合わせた結果のa
(c) aの10の位の数値
(d) aの1の位の数値
(e) aの10の位の数値とaの1の位の数値を加算した値b
(f) n＋mとbが等しいかどうか
(g) n，mの組み合わせをすべて表示

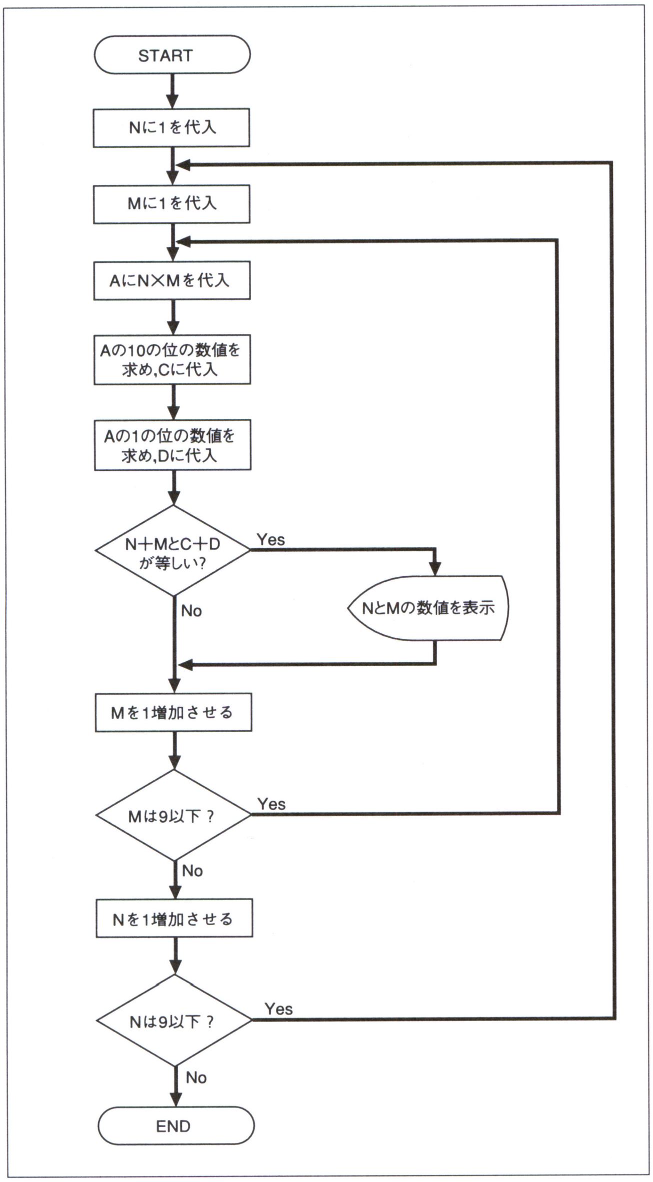
というような要素が含まれることがわかる。問題となっている処理を実現するためには，これらの要素を何らかの形で組み込む必要があり，フローチャートにもこれらの要素がすべて含まれていなければならない（**第2図**）。

フローチャートを見てみると，これらの要素のうち，a，c，dが明確に示されている。よって，空欄は他の要素に対応する可能性が高い。この点に注意しながら，個々の空欄を調べていくことにする。

最初の空欄**ア**については，アの内容が変数Aに代入されることと，代入後に10の位の数値と1の位の数値をそれぞれ取り出していることから，nとmを掛け合わせた結果に対応していることになる。よって，**(7)のN×M**が対応する。こうすることで，問題文の「nとmを掛け合わせた結果のaについて，10の位の数値と1の位の数値」まではフローチャートに含まれるようになる。

次の空欄イについては条件判断に含まれており，条件が成立するとNとMを表示していることがポイントになる。表示が行われるということは，問題文におけるn＋mとbが等しいということになるため，この条件判断で利用される数値は，n＋mとbに対応している可能性が極めて高い。フローチャートにおけるCとDはそれぞれ10の位の数値と1の位の数値に対応していることから，これを等しいかどうかを判断する相手としては，n＋mに対応するN＋Mが妥当である。よって，**イは(5)のN＋M**となる。これによって，e，f，gの内容が含まれるようになり，要素の面ではフローチャートは完成する。

空欄ウと空欄エについては，ループを継続する条件に対応する。「Mを1増加させる」処理と，「Nを1増加させる」処理が示されていることから，初期値の1からカウントアップしていることになる。問題文から，nとmの範囲は1桁の自然数となっ



《第3図》完成されたフローチャート

ているため，ループは1から9まで行う必要があり，ループ継続条件は9以下となる。よって，**ウ，エとも(3)の9**が対応する。

ア：7
イ：5
ウ：3
エ：3

完成させたフローチャートを**第3図**に示す。フローチャートの動作をみると，NとMを1から9まで変化させながら，問題の条件を満たすかどうかを調べ，満たす場合に表示を行っていることが確認できる。

基本的にフローチャートはプログラムと1対1に対応するものなので，プログラムの空欄を埋めるタイプの問題と同じように，必要な要素が存在するかどうかといった点に着目すれば解答が見つかる。見慣れないフローチャートが出たからといって慌てないようにしたい。

設問2では，設問1で完成させたフローチャートから実際のプログラムを完成させることが求められている。フローチャートはプログラムと対応したものであるため，フローチャートの要素がプログラムに組み込まれているかどうかを確認していく。

まず，NとMを1から9まで変化させることについては，10行，20行，80行，90行の2重FOR−NEXTループによって実現されている。次に，N×Mを計算し，10の位の数値と1の位の数値を求める部分であるが，この部分でプログラムが完成しているのは40行だけである。40行の動作は，Aを10で割り，整数部分を取り出している（厳密には，Aを10で割り，それを越えない最大の整数を求めている）。よって，10の位の数値を計算している可能性が高い。となれば，30行はN×Mを計算していることになり，**オは(1)のN＊M**である可能性が高い。

60行では，条件判断が行われ，条件が成立した場合には70行のPRINT文をキャンセルするようになっている。条件判断はこの行にしか存在しないことから，この部分で問題文におけるn＋mとbが等しいかどうかを判断しているのは確実となる。そうであれば，50行は1の位を計算している可能性が高い。1の位の計算のためには，10で割った余りを求める必要があることから，**カの部分は(4)のC＊10**になっている可能性が高い。

Cが10の位の数値，Dが1の位の数値であれば，**キは(3)の<>**とし，一致しない場合に70行のPRINT文をキャンセルするようにしなければならない。

これらの結果を組み込むと，プログラムは以下のようになる。

```
10 FOR N=1 TO 9
20  FOR M=1 TO 9
30   A=N*M
40   C=INT(A/10)
50   D=A-C*10
60   IF N+M<>C+D THEN GOTO 80
70   PRINT "n=";N;":m=";M
80  NEXT
90 NEXT
100 END
```

一部の選択肢は可能性が高いということで解答しているため，このプログラムの動作を確認して裏付けを取る。30行でNとMの積を計算し，40行でCに10で割った結果を，50行でDに10で割った余りをそれぞれ代入している。この結果，CはN×Mの10の位の数値に，DはN×Mの1の位の数値になる。そして，60行でN＋MとC＋Dが一致しない場合に70行をキャンセルすることで，一致する組み合わせだけを70行で表示させている。よって，問題が求めている組み合わせを表示することができる。

オ：1
カ：4
キ：3

空欄を埋めるタイプの問題については，まず最初に確実に埋

《第1表》積の10の位の数値と1の位の数値の和の表

	2	3	4	5	6	7	8	9	m
2	4	6	8	10	12	14	16	18	n×m
	4	6	8	1	3	5	7	9	b
3		9	12	15	18	21	24	27	
		9	3	6	9	3	6	9	
4			16	20	24	28	32	36	
			7	2	6	10	5	9	
5				25	30	35	40	45	
				7	3	8	4	9	
6					36	42	48	54	
					9	6	12	9	
7						49	56	63	
						13	11	9	
8							64	72	
							10	9	
9								81	
								9	

n

めることができる空欄を埋めておき，その後予想を立てて動作を調べながら残りの空欄を埋めていくという手順になる。

設問３では，プログラムを実際に動作させた場合の結果が求められている。実行終了までの表示行数が問われていることから，全部の組み合わせについて確認を行う。ただし，プログラムでは９×９の81とおりの組み合わせを調べているが，ｎとｍは入れ替えが可能なので，実際には45とおりとなる。また，片方が１の場合についても条件を満たせないため，残りは36とおりとなる。これらについて，確認を行う。

第1表のような表を作成して確認してみると，

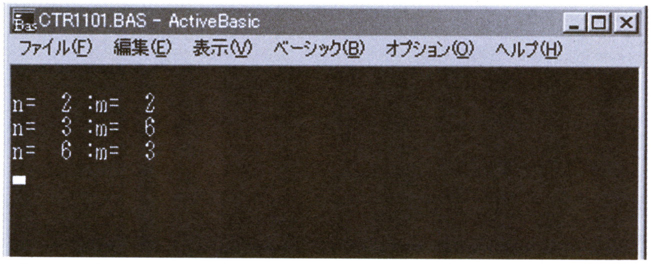
ｎ＝２：ｍ＝２

ｎ＝３：ｍ＝６

ｎ＝６：ｍ＝３（ｎ＝３：ｍ＝６の入れ替え）

の３とおりについて条件が成立することがわかる。よって，**コサは03**となる。また，３行目に表示される内容については，二つのFORループのうち，変数Ｍに関するのループが内側に入っており，変数Ｎが外側になっていることから，ｎが小さいものから順番に表示されていくことが判る。よって，３行目に表示されるのは，最もｎが大きい「ｎ＝６：ｍ＝３」の組み合わせとなり，**クは６，ケは３**となる。

ク：６
ケ：３
コ：０
サ：３



《写真１》プログラムを実行した結果。２－２，３－６，６－３の３とおりが表示される

実際にプログラムを実行した結果を**写真１**に示す。３行の結果が表示されており，３行目は「ｎ＝６：ｍ＝３」になっていることがわかる。

プログラムの実行結果を求めるタイプの問題については，計算にかかる時間がポイントとなる。あまりにも時間がかかるようであれば，数式を使って解析的に解く方法を考えたり，他の問題を優先する方が有利である。ループ回数が多い場合は要注意である。

予想問題4

整数ａ，ｘ，ｙについて，ｘ＞０，ｙ＜０とした場合に，$y=x^2-ax+3$を満たす，ｘ，ｙの組み合わせを調べたい。このために以下のようなプログラム１を作成してみた。

プログラム１

```
10 INPUT A
20 X=1
30 Y=X*X-A*X+3
40 IF Y>=0 THEN GOTO 80
50 PRINT "X=";x;"  Y=";y
60 X=X+1
70 GOTO 30
80 END
```

［１］プログラム１を実行し，Ａとして５を入力した場合，50行はア回実行され，最後に表示されるのはＸ＝イ，Ｙ＝－ウである。また，Ａとして４を入力した場合，30行はエ回実行される。

［２］$y=x^2-4x+3$について，ｘ＞０，ｙ＜０を満たすｘとｙの組み合わせはオカとおりである。プログラム１が正しく動作するように，プログラム１を修正し，プログラム２を作成した。プログラム２の空欄に適切な選択肢を選べ。

プログラム２

```
10 INPUT A
20 X=1
21 Y=X*X-A*X+3
22 IF Yキ0 THEN GOTO 50
23 IF X*2-Aク0 THEN GOTO 80
24 X=X+ケ
25 GOTO 21
30 Y=X*X-A*X+3
40 IF Y>=0 THEN GOTO 80
50 PRINT "X=";x;"  Y=";y
60 X=X+1
70 GOTO 30
80 END
```

キとクに関する選択肢

(1) ＞　(2) ＜　(3) ＜＞　(4) ＝

ケに関する選択肢

(1) ＋1　(2) ＋2　(3) ＋3　(4) −1　(5) −2　(6) −3
(7) ＋A　(8) −A

［3］プログラム2を実行し，Aとして4を入力した場合，23行のIF文はコサ回実行される。また，Aとして32を入力した場合，50行のPRINT文はシスセ回実行され，表示されるYの値のなかで最も小さい値はソである。

ソに関する選択肢

(1) −5　(2) −31　(3) −64　(4) −124　(5) −253
(6) −378　(7) −515　(8) −788　(9) −1024

解法4

センター試験のBASIC問題では，基本的にプログラム，あるいはフローチャートは一つだけで，それに関する設問が用意されることが多い。しかし，問題のプログラムなどを修正させ，複数のプログラムが扱われるケースもある。とくに，プログラムを修正していくタイプの問題は，実際のプログラミングで行われるデバッグと似た対応が求められるため，現実的な問題と言える。

この問題の場合も，設問を一とおり読むことで，最初に用意されたプログラムが不十分であり，プログラムの修正が行われていることがわかる。よって，最初のプログラムは問題の処理内容を正しく実行できていない可能性がある。これを踏まえて，設問に対応していく。

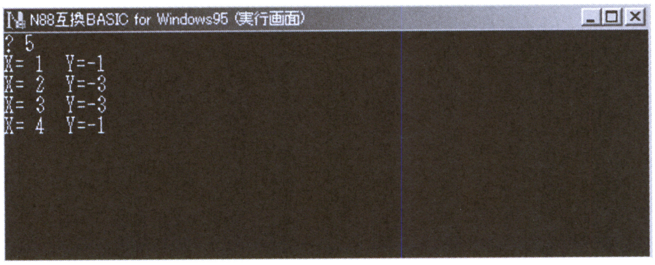
最初の設問では，プログラム1を実行した場合の結果が問われている。問題自体は数学的に解くことも十分可能だが，プログラム1は問題に示されたとおりに動作しない可能性があるため，純粋にプログラム1の動作をシミュレーションすることにする。

プログラム1は，70行のGOTO文によるループが構成されており，40行のIF文が成立することでループから抜け出す構成となっている。Aとして5を入力した場合の動作を変数Xの変化に着目してシミュレーションすると以下のようになる。

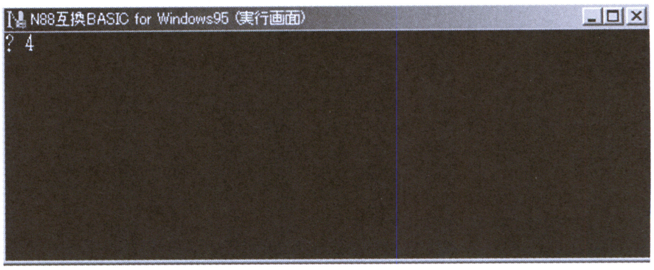
Xの値	Yの値	40行のIF文	50行の表示
1	−1	不成立	X=1 Y=−1
2	−3	不成立	X=2 Y=−3
3	−3	不成立	X=3 Y=−3
4	−1	不成立	X=4 Y=−1
5	3	成立	表示されない

この結果から，50行は4回実行され，最後に表示されるのはX=4の場合の「X=4 Y=−1」である。

ア：4



《写真2》問題のプログラム1を実行し，5を入力した場合。四つの組み合わせが表示されている。



《写真3》問題のプログラム1を実行し，4を入力した場合。何も表示されないままプログラムの動作が終了する。

イ：4
ウ：1

Aとして4を入力した場合についても同様のシミュレーションを行う。

Xの値	Yの値	40行のIF文	50行の表示
1	0	成立	表示されない

1回目の計算でYの値は0となってしまうため，30行は1回実行されただけでプログラムが終了する。

エ：1

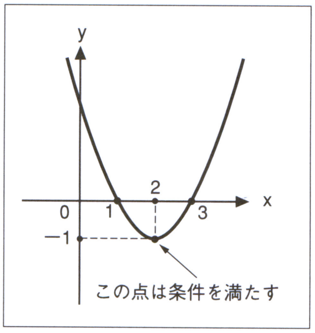
実際にプログラム1を実行した結果を**写真2**と**写真3**に示す。A=5とすると，4行の結果が表示され，最後の行は「X=4 Y=−1」となっている。また，A=4とすると，結果が表示されない。

プログラムのシミュレーションを行う場合は，ループ変数となっている変数を基準とし（プログラム1の場合は変数X），その変数の変化に応じて動作がどのように変化していくかを追跡するのが一般的な方法である。ループするにつれて変化する変数（プログラム1の場合は変数Y）をしっかりと抑えて抜けのないようにする必要がある。

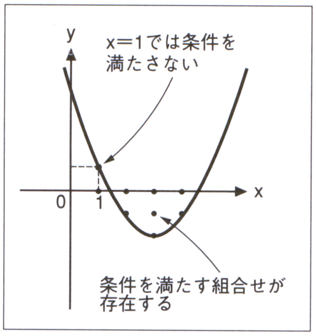
設問2は，BASICプログラムから離れた数学的な設問と，それに関係するプログラム修正の設問からなる。数学的な設問では，

$$y=x^2-4x+3 \quad \cdots (1)$$

について，$x>0$と$y<0$を満たす組み合わせを数える必要がある。このような設問の場合，グラフを作成するのが簡単であ



《第4図》式1のグラフ



《第5図》プログラム1が誤作動する場合

る。式１を２とおりに変形し，

y＝(x−2)^2−1　・・・（2）
y＝(x−1)(x−3)　・・・（3）

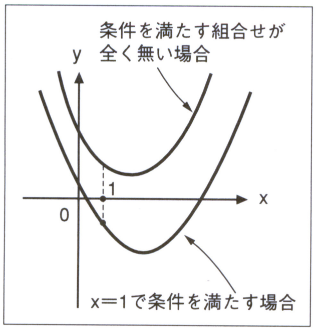
とする。式２からこの式の最小値とその時のｘの値が，式３からＸ軸と交わる位置がそれぞれ明らかになる。これによって，この式が**第4図**のようなグラフで表されることがわかる。このグラフを見ると，条件を満たす組み合わせは「X＝2 Y＝−1」の１とおりしかないことがわかる。

オカ：０１

設問１でＡを４とした場合は，何も表示されないままプログラム１は終了してしまうが，実際には「X＝2 Y＝−1」という組み合わせがあり，これを正しく表示できていないことが判明する。よって，プログラム１にはミスがあり，問題の求めている動作を実現できていないことになる。

設問２の後半では，このミスを修正し，プログラム２を完成させることが求められている。プログラム２をプログラム１と比較してみると，21行から25行が追加されていることがわかる。Ａの入力と変数Ｘの初期化を除けば，プログラム１の前にプログラムを追加した形になる。よって，プログラム１を実行する前に，何らかの処理を行っているという捉え方ができる。

プログラム１が正常に動作しないのは，**第5図**のように条件を満たすＸとＹの組み合わせが存在するが，X＝1では条件を満たさない場合である。**第6図**のように条件を満たす組み合わせがない場合や，X＝1の時点で条件を満たす場合は正常に動作するが，そうでない場合はX＝1の段階で40行のIF文が成立してしまうことから，組み合わせの探索が行われない。



《第6図》プログラム1が正常な結果を返す場合

このことを，プログラム２の追加部分がおおよそプログラム１の前処理に該当することと組み合わせて考えてみると，**第5図**のような場合において，追加部分で条件を満たすＸになるまでＸを増加させていると考えることができる。つまり，プログラム１では常にX＝1から開始されていた探索処理を，条件が成立し始めるＸから始めるように変更していると考えることができる。

この点を考慮しながら空欄を調べていく。最初の空欄のキは，IF文の成立条件に関係している。条件が成立すると50行にジャンプするが，50行は問題の条件を満たすＸとＹの表示を行っており，ジャンプしたからには条件が成立していなければならない。よって，Ｙの成立条件である，Y＜0が成立している必要があり，**キは(2)の＜**となる。

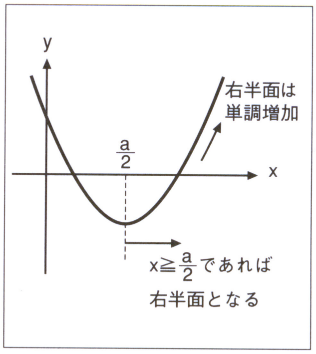
クもIF文の成立条件に関係している。条件成立時のジャンプ先がEND文の80行になっているため，条件が成立するとプログラムが終了することになる。25行のGOTO文によって，21行から25行はループになっており，このループ内では問題の条件は成立しない。このため，23行が成立するのは，問題の条件を満たす組み合わせが一つもない場合であり，そのことを判断していることになる。Ｘの範囲はX＞0であるため，探索すべき組み合わせは無数に存在する。よって，何らかの方法で，これ以上探索しても条件を満たす組み合わせが存在しないことを確認していることになる。Ｘの探索は，21行から25行のループの前の20行でＸが１に初期化されていることから，１からプラス方向へ探索していることが確実である。よって，これ以上Ｘを増やしても条件を満たす組み合わせが存在しない，ということを23行で判断していることになる。グラフの形は**第5図**のような放物線であることから，Ｙが０以上で，かつＸの値が放物線の右半分に対応している場合，それ以上Ｘを増やしてもＹが０未満になることはあり得ない。これは，Ｘが放物線の頂点のＸ座標の値以上になっていた場合が該当する。

放物線の頂点のＸ座標は，

y=x^2−ax＋3　・・・（4）
y＝$(x-\frac{a}{2})^2+3-\frac{a^2}{4}$　・・・（5）

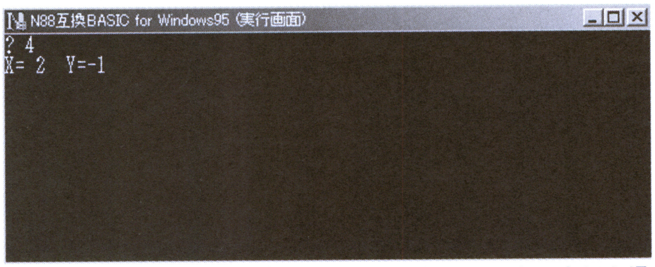
から，A/2になることがわかる。よって，ＸがA/2以上になれば，放物線の右半面の領域に入っていることになる（**第7図**）。23行の条件式は「X＊2−Aク0」となっていることから，**ク**として当てはめることができるのは**(1)の＞**だけである。＞＝でも正しい動作になるが，候補に含まれていないため，＞が解答となる。

ケについては，探索を行うＸの値を変化させていることになる。方向としてはプラス方向であることが確実である。また，変化幅は１でなければ条件が成立するＸの組み合わせを見落とす可能性がある。よって，**ケは(1)の＋1**となる。



《第7図》放物線の右半面

キ：２
ク：１
ケ：１



《写真4》修正されたプログラム2を実行し，4を入力した場合。「X=2 Y=−1」の組み合わせが表示されている。

空欄を埋めたプログラム2を実行し，A＝4とした場合は**写真4**のように「X=2 Y=−1」の組み合わせが表示される。

設問3の前半ではプログラム2を実行した場合に，23行のIF文が何回実行されるかが問われている。(3) 式から，Y<0となっているXの範囲は狭いことがわかるため，単純にシミュレーションを行う。

空欄を埋めたプログラム2は以下のようになる。

```
10 INPUT A
20 X=1
21 Y=X*X-A*X+3
22 IF Y<0 THEN GOTO 50
23 IF X*2-A>0 THEN GOTO 80
24 X=X+1
25 GOTO 21
30 Y=X*X-A*X+3
40 IF Y>=0 THEN GOTO 80
50 PRINT "X=";x;"  Y=";y
60 X=X+1
70 GOTO 30
80 END
```

A=4の場合について，ループ変数Xを基準とし，22行と23行のIF文の動作を調べていく。

Xの値	Yの値	22行のIF文	23行のIF文
1	0	不成立	不成立
2	−1	成立	実行されない

この結果から，23行のIF文は1回だけ実行されることになり，**コサは01**となる。

設問3の後半では，A=32の場合について，50行のIF文の実行回数と，表示されるYの最小値が求められている。A=32となっていることから，(3) 式より条件を満たす可能性のあるXの範囲が広くなっている。よって，単純にシミュレーションを行うのは時間の面で問題があると予想できる。そこで，シミュレーションを行わず，解析的に解答を求めることにする。

プログラム2の動作内容から，50行のPRINT文の実行回数は，問題の条件を満たす組み合わせの数に等しい。また，表示される最も小さなYの値は，A=32とした時の条件式の最小値に等しい。

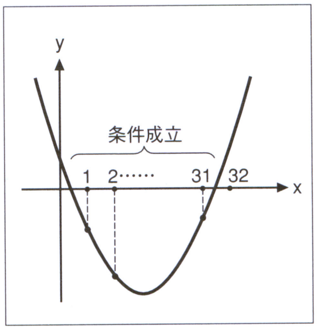
$$y=x^2-32x+3 \quad \cdot\cdot\cdot \ (6)$$

(6) 式の解は，

$$x=\frac{-32\pm\sqrt{32^2-4.1-3}}{2.1}$$

$$=-16\pm\sqrt{253}$$

となる。ここで，この解の値について調べると，$16^2>253>15^2$より，解の一つは0と1の間に，もう一つは31と32の間に存在することが判る。グラフにすると**第8図**のようになり，二つの解の間のXについてすべて問題の条件が成立する。よって，条件が成立するXの個数は1から31の31個となり，50行のPRINT文も31回実行されることになる。よって，**シスセは031**となる。



《第8図》式6のグラフ

また，(5) 式より，最小値は

$$ymin=3-\frac{32^2}{4}$$

$$=-253$$

となり，**ソは(5)の−253**となる。

コ：0
サ：1
シ：0
ス：3
セ：1
ソ：5

最後に

今回は，センター試験のBASIC問題として比較的新しいタイプであるフローチャート対応タイプと，プログラム修正タイプを取り上げた。BASIC問題は実際にパソコンが利用できないという制約の影響で問題のバリエーションに限界があるが，これらの二つのタイプはその中で工夫されて登場してきた問題である。どちらも出題実績があることから，今後も出題される可能性があるため，要注意である。

センター試験までの期間は，体調を整えるだけでなく，精神的に安定させることも重要である。冷静に落ち着いた状況で試験に臨めば，不注意による失敗などが起こりにくく，安定した力を発揮できる。これまでの自分の努力を振り返り，やれるだけのことはやった，と自信をもって試験に臨んで欲しい。朗報を期待している。

パソコン認定試験　対策講座

連載　第4回
表計算ソフトウエア

パソコン認定試験主催：社団法人パーソナルコンピュータユーザ利用技術協会

パソコン認定試験にチャレンジ！

“パソコンできます”と言っても，本当の実力がどのくらいか認めてもらえません。そのような技術力を認めさせるために，一度，いま注目されているパソコン認定試験に挑戦してみませんか。

1983年に始まり，受験者総数は，すでに56万人，合格者総数も31万人を越えました。今では企業の採用試験の目安として使われるなど，時代のニーズにあった資格です。試験レベルは4段階，一般的な事務処理なら3級・4級の資格で十分と言えます。

とくに，パソコン認定試験合格者は，文部科学省が実施する大学入学資格検定の受検科目（情報関係基礎）が免除されます。また，学生またはキャリアアップしたい社会人にまで，パソコンの利用技術の資格試験に挑戦する傾向が急速に高まり，企業側も新規採用の条件に，また職員の昇格，奨励金の面で有資格者に優遇処置を図るところが多く，評価の対象となっています。

今回は，表計算ソフトウェアに関する，次のような例題を掲載し，解説しました。

例題１　表計算ソフトウェアに関する記述について，次の(1)～(10)に入れるべき最も適切な字句を解答群の中から選べ。

	A	B	C	D	E	F	G	H	I	J
2										
3			叡智大学2次試験成績							
4										
5	受験番号	氏名	数学	物理	英語	数学偏差値	物理偏差値	英語偏差値	総合点	最低点
6	1	山田健一	88	90	60	51.75	51.76	38.85	327.0	60
7	2	佐藤太郎	76	80	66	42.47	40.63	42.35	300.0	66
8	3	高橋翔	56	70	60	27.01	29.51	38.85	249.0	56
9	4	青山かの子	96	98	90	57.93	60.66	56.36	381.0	90
10	5	谷山宏	100	100	100	61.03	62.88	62.19	400.0	100
11	6	加藤浩子	66	73	100	34.74	32.85	62.19	308.5	66
12	7	吉岡泰子	98	96	86	59.48	58.43	54.02	377.0	86
13	8	谷川恵	95	91	61	57.16	52.87	39.43	340.0	61
14	9	木下次郎	92	93	95	54.84	55.09	59.28	372.5	92
15	10	近藤豪	78	88	85	44.02	49.53	53.44	334.0	78
16	11	近松久江	91	82	70	54.07	42.86	44.69	329.5	70
17	12	津田松子	88	80	100	51.75	40.63	62.19	352.0	80
18	13	斎藤幸	99	100	90	60.25	62.88	56.36	388.5	90
19	14	青空健太	85	95	80	49.43	57.32	50.52	350.0	80
20	15	飯島毅	70	88	85	37.83	49.53	53.44	322.0	70
21	16	吉田賢治	100	100	38	61.03	62.88	26.01	338.0	38
22	17	赤木真紀	95	90	92	57.16	51.76	57.52	369.5	90
23	18	進藤肇	70	85	65	37.83	46.19	41.77	297.5	65
24	19	空野悟	86	81	80	50.20	41.74	50.52	330.5	80
25										
26										
27		平均	85.74	88.42	79.11					
28		標準偏差	12.94	8.99	17.14					
29		数学物理の相関係数		0.80						

図1.1

図1.1は，叡智大学ロボット学科の2次入学試験の数学，物理及び英語の成績表である。成績は各科目とも100点満点であり，総合点は数学及び物理の得点は5割増しとする。合格は上位10名とする。ただし，どれかの科目で調整する前の点が40点以下のものがあれば例え総合点がよくても不合格とする。

各科目ごとの平均点及び標準偏差を計算する。次に，数学の成績と物理の成績との相関係数を求める。各受験生について科目の偏差値，総合点及び最低点を計算する。その後，総合点によって並び換えた表図1.2を作り，最低点が40点を超えるものの上位10名を合格とする。計算の手順を次にあげる。

このソフトウェアでは，$を行又は列を表す記号の前に付けると，計算式を複写してもその行又は列が変わらないことを表している。ただし，この表計算ソフトウェアでは，式は“＝”で始まり，関数は次のとおりになっているものとする。

平均	AVERAGE(範囲)	標準偏差	STDEV(範囲)	相関係数	CORREL((範囲),(範囲))
和	SUM(範囲)	最小	MIN(範囲)		

セルXからセルYへの範囲指定は，X:Yで行う。

各人の偏差値は，次の式で計算される。

Aさんのある科目の偏差値

＝(Aさんのその科目の成績－受験生のその科目の成績の平均値)＊10/受験生のその科目の成績の標準偏差＋50

手順１．科目ごとの平均点をそれぞれC27，D27及びE27に求める。セルポインタをセルC27に移動してから，＝AVERAGE((1))と入力し，この計算式を“複写”によってセル (2) からE27まで複写する。

手順２．科目ごとの標準偏差を求める。数学，物理及び英語の標準偏差をC28，D28及びE28にそれぞれ求めるために，セルポインタをC28に移動してから，＝ (3) (C6:C24)と入力し，この計算式を“複写”によってセルD28から (4) まで複写する。

手順３．個人ごとの偏差値を科目ごとに求める。セルポインタをセルF6に移し，＝ (5) ＊10/C＄28＋50と入力する。次にこの計算式を“複写”によってセルF7からF24まで複写する。更に，この計算式を“複写”によってセル (6) からG24及びH6からH24まで複写する。

手順４．総合点を計算する。セルポインタをセルI6に移し， (7) を入力する。次に，この計算式を“複写”によってセルI7からI24まで複写する。

手順５．最低点を計算する。セルポインタをセルJ6に移し，＝ (8) を入力する。次に，この計算式を“複写”によってセルJ7から (9) まで複写する。

手順６．セルD29に数学の成績と物理の成績との相関係数を求めるため，セルD29に式＝ (10) (D6:D24))を入力する。

総合得点順に並び換えて図1.2が得られる。これから最低点が40点を超えるものの上位10人を合格とした。

	A	B	C	D	E	F	G	H	I	J
2										
3			叡智大学2次試験成績							
4										
5	受験番号	氏名	数学	物理	英語	数学偏差値	物理偏差値	英語偏差値	総合点	最低点
6	5	谷山宏	100	100	100	61.03	62.88	62.19	400.0	100
7	13	斎藤幸	99	100	90	60.25	62.88	56.36	388.5	90
8	4	青山かの子	96	98	90	57.93	60.66	56.36	381.0	90
9	7	吉岡泰子	98	96	86	59.48	58.43	54.02	377.0	86
10	9	木下次郎	92	93	95	54.84	55.09	59.28	372.5	92
11	17	赤木真紀	95	90	92	57.16	51.76	57.52	369.5	90
12	12	津田松子	88	80	100	51.75	40.63	62.19	352.0	80
13	14	青空健太	85	95	80	49.43	57.32	50.52	350.0	80
14	8	谷川恵	95	91	61	57.16	52.87	39.43	340.0	61
15	16	吉田賢治	100	100	38	61.03	62.88	26.01	338.0	38
16	10	近藤豪	78	88	85	44.02	49.53	53.44	334.0	78
17	19	空野悟	86	81	80	50.20	41.74	50.52	330.5	80
18	11	近松久江	91	82	70	54.07	42.86	44.69	329.5	70
19	1	山田健一	88	90	60	51.75	51.76	38.85	327.0	60
20	15	飯島毅	70	88	85	37.83	49.53	53.44	322.0	70
21	6	加藤浩子	66	73	100	34.74	32.85	62.19	308.5	66
22	2	佐藤太郎	76	80	66	42.47	40.63	42.35	300.0	66
23	18	進藤肇	70	85	65	37.83	46.19	41.77	297.5	65
24	3	高橋翔	56	70	60	27.01	29.51	38.85	249.0	56
25										
26										
27		平均	85.74	88.42	79.11					
28		標準偏差	12.94	8.99	17.14					
29		数学物理の相関係数		0.79						

図1.2

解答群

01 STDEV	02 SUM	03 AVERAGE	04 CORREL
05 MEAN	06 C4:C24	07 C4:E4	08 C6:C24
09 (C6-C＄27)	10 (C＄6-＄C27)	11 CORREL((C6:C24),	12 D27
13 E28	14 C28	15 H24	16 G24
17 MIN	18 MAX	19 MIN(C6:E6)	20 I24
21 G6	22 J24	23 ＝(C6＋D6)＊1.5＋E6	

［第38回(2002年7月実施)パソコン認定試験４級より］

例題１の解説

表計算ソフトウェアで成績集計表を作成するための基本操作に関する出題です。標準偏差や相関係数など普段なじみのない関数もありますが，関数の名前や使用方法が問題文に書かれていますので，対象となるデータの範囲やセル参照の方法を間違えなければ解くことができます。

表計算ソフトウェアは，表形式でデータを集計し加工することができるソフトウェアです。家計簿，成績表，予算書などのほか，各種の帳票に利用できます。表計算ソフトウェアでは，画面を集計用紙に見立て，行方向と列方向に番号を付けています。行と列の交差した「ます目」をセルといいます。セルの位置指定の方法は，列番号と行番号を組み合わせて行います。例えば，問題の図1.1で「受験番号」という文字データの入っているセルはA5です。

表計算ソフトウェアの主な機能としては，集計，比較，関数などの計算機能，挿入，削除などの編集機能，並べ替えなどの分類機能，グラフの作成機能，印刷などの出力機能，データの格納機能などがあります。

手順１　科目ごとの平均点の計算です。平均を算出する便利な関数がAVERAGEです。そこで，数学の平均点は“C27＝AVERAGE(C6:C24)”となります。具体的には，セルC27をアクティブ（セルポインタを移動させること）にして“＝”以下の入力を行います。さて，物理と英語の平均点も列が異なるだけで計算方法は同じです。表計算ソフトウェアではこうした場合，一箇所だけ計算のルールを設定すれば，残りの箇所は計算ルールの複写ができます。そして，複写の際は特に指定しない限り，セルの参照は相対的に行われることになっています。つまり，C27の計算式をD27とE27に複写すると，D27にはAVERAGE(D6:D24)が，E27にはAVERAGE(E6:E24)が設定できるのです。［(1)］には「08　C6:C24」が，［(2)］には「12　D27」が入ります。

手順２　科目ごとの標準偏差の計算です。標準偏差を算出する関数がSTDEVです。数学の標準偏差の算出は“C28＝STDEV(C6:C24)”となります。そこで，まず数学の標準偏差の計算式を入力し，それを物理の標準偏差（D28）と英語の標準偏差（E28）に複写します。［(3)］には「01　STDEV」が，［(4)］には「13　E28」が入ります。

手順３　個人ごとの科目別偏差値の計算です。偏差値の計算は問題文に書かれていますので，それに従って計算式を作ります。受験番号1の数学の偏差値は，“F6＝(C6-C27)＊10/C28＋50”となります。これをこのまま複写して物理の偏差値とすると，“G6＝(D6-D27)＊10/D28＋50”となり，英語に複写しても正しく算出ができます。ところが，受験番号1の数学の偏差値を受験番号2に複写してみると，“F7＝(C7-C28)＊10/C29＋50”となりF7のセルには“#DIV/0!”などの表示がされてしまいます。計算式を見直してみると，数学の平均点のセルC27がC28に，数学の標準偏差C28がC29になっていることがわかります。計算式の複写の際，セルの参照が相対的に行われたので，当然のことといえます。しかし，数学の平均点（C27）や標準偏差（C28）はどの受験生からみても不変のセルです。こうした場合，複写によってセルが変化しないよう「絶対参照」を指示します。具体的には，複写により変化してはいけない列又は行の前に＄を付けて表記します。そこで，“F6＝(C6-C＄27)＊10/C＄28＋50”としてから複写すると，平均点の行番号27と標準偏差の行番号28が固定できます。［(5)］には「09　(C6-C＄27)」が，［(6)］には「21　G6」が入ります。

手順４　総合点の計算です。数学，物理，英語の3科目の合計ですが，数学と物理は5割増しと問題文にあるので，英語は100点，数学は150点，物理は150点の400点満点です。そこで，受験番号1の総合点を“I6＝(C6＋D6)＊1.5＋E6”と入力してから他の受験生に計算式の複写を行います。［(7)］には「23　＝(C6＋D6)＊1.5＋E6」が入ります。

手順５　最低点は最小値を求める関数MINを利用します。ここでは個人ごとに3科目の最低点を表示するので，受験番号1の最低点は“J6＝MIN(C6:E6)”となります。これを他の受験生に複写します。［(8)］には「19　MIN(C6:E6)」が，［(9)］には「22　J24」が入ります。

手順６　相関係数を求める関数はCORRELです。ここでは数学と物理の相関ですから，数学のデータ系列と物理のデータ系列を指定します。つまり，“D29＝CORREL((C6:C24),(D6:D24))”とすればいいので，［(10)］には「11　CORREL((C6:C24),」が入ります。

例題2　表計算ソフトウェアを利用して，請求書を自動的に作成する手順に関する次の記述中の(1)～(10)に入れるべき最も適切な字句を解答群の中から選べ。

請求書は，企業などにおいて顧客との取引に対応して日常的に作成する必要があることから，自動的に作成できることが望ましい。そのために，図2.1に示す売上げ例に関するデータ（売上表）に基づいて，図2.2に示す顧客別に売上げの内訳を示し，合計金額（ピボット表）を計算する。次に，これらのデータから顧客別の請求書を自動的に発行することを考える。使用する表計算ソフトウェアでは，マクロプログラム機能を備え，また，処理手順を手続きとして記述するものとする。ここでは，日付の表示を省略する。

	A	B	C
1	顧　客	品　名	金　額
2	（社）パソコン技術協会	パソコン Set A	298,000
3	東京国際（株）	プリンタ Model 1	79,800
4	東京国際（株）	イメージスキャナ α	120,000
5	（株）日本コンピュータ	パソコン Set A	298,000
6	アジア貿易（株）	パソコン Set B3	78,000
7	アジア貿易（株）	プリンタ Model 2	112,500

図2.1　作業用紙1　売上表

	A	B	C	D	E	F	G
1		品 名					
2	顧客	イメージスキャナ α	パソコン Set A	パソコン Set B	プリンタ Model 1	プリンタ Model 2	総 計
3	（株）日本コンピュータ		298,000				298,000
4	（社）パソコン技術協会		298,000				298,000
5	アジア貿易（株）			378,000		112,500	490,500
6	東京国際（株）	120,000			79,800		199,800
7	総計	120,000	596,000	378,000	79,800	112,500	1,286,300

図2.2　作業用紙2　ピボット表

マクロプログラムの記述にあたっては，作業用紙に配置できる表計算ソフトウェアのファイル，それを構成する作業用紙，作業用紙中のセル，セル範囲，ボタン，図形などはオブジェクトと呼ばれ，各オブジェクトは属性をもつとともに，オブジェクトに対して実行できる操作であるメソッドをもつ。

使用する表計算ソフトウェアでは，表2.1に示す宣言，オブジェクト，属性及びメソッドが使用できるものとする。

表2.1　宣言，オブジェクト，属性及びメソッドの機能

（宣言）	
Option Explicit	手続き内のすべての変数を明示的に宣言する。
Private	手続き内の局所変数を指定する。
（オブジェクト）	
Application	使用する表計算ソフトウェアを表す。
Range	セル又はセル範囲を表す。
（属性）	
ActiveWindow	対象とする窓（ウィンドウ）を指定する。
Cells(行，列)	表の行及び列で示すセル範囲を指定する。
Column	対象セル範囲の最初の列の番号を指定する。
End	対象セルを含む領域の終端セルを指定する。
Name	ファイル名又はフォルダ名を指定する。
Range	セル又はセル範囲を指定する。
Row	対象セル範囲の先頭の行番号を指定する。
ScreenUpdating	Trueの場合，マクロの実行中に画面表示を更新する。そうでない場合は，何もしない。
SelectedSheets	選択されたすべての作業用紙を指定する。

（メソッド）	
ClearContents	選択範囲を初期化する。
Select	オブジェクトを選択する。
Sheets	指定したファイルに含まれる一つの作業用紙又は作業用紙の集合を返す。

更に，使用するマクロプログラムでは，次の組込み定数及び関数が用意されているものとする。

表2.2　組込み定数及び関数の機能

（組込み定数）	
xlDatabase	使用する表計算ソフトウェアのデータベース
xlDataField	データ領域
xlDown	下へ移動
xlToRight	右に移動
（関数）	
Round	小数を含む数値を整数にまるめる。

手続きにおいて，Dimは手続きで使用される変数の宣言，Privateはサブルーチンの手続きに関する局所変数を表す。更に，Private Subは，他のモジュールからは実行できない手続きの宣言を表す。

次に，請求書作成の手順を示すが，ここでは図2.1に示す売上表は，作業用紙1として既に作成されているものとする。

請求書作成の手順：

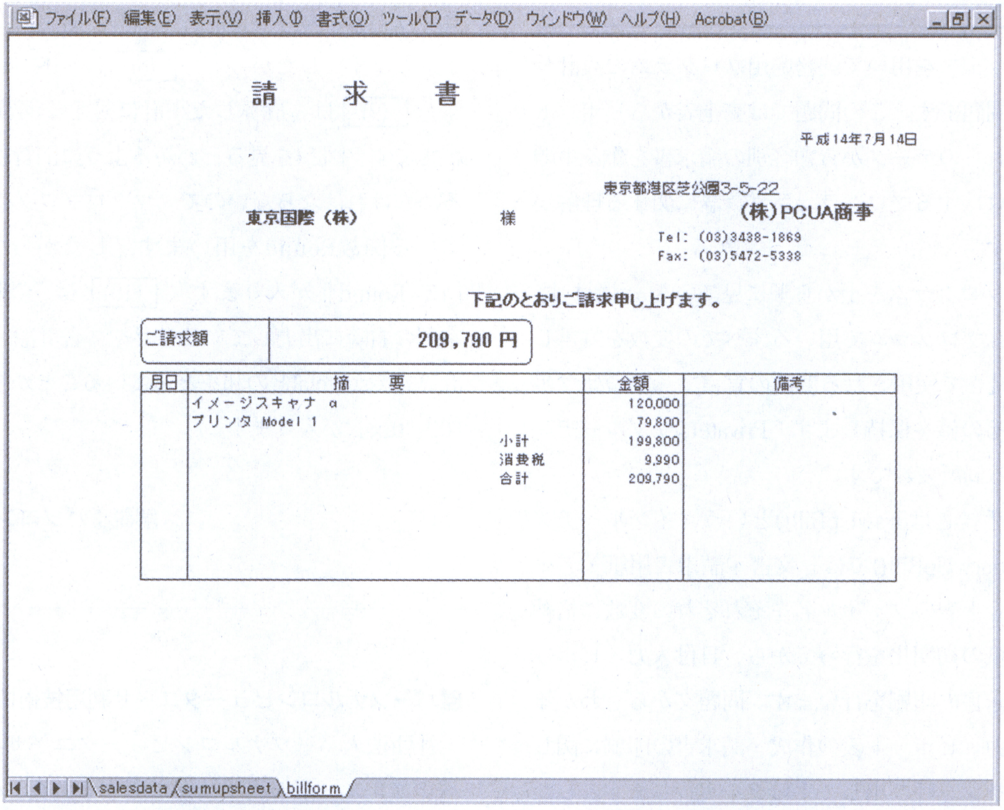
請求書の作成を行うには，図2.3に示すマクロプログラムに記述された手続きを実行する。まず，メインルーチンの手続きdrawUpBill()が実行されると，サブルーチンinputDate()によって請求書の日付の入力を受け付ける。次に，図2.2に示されるピボット表がサブルーチンcreateSumupSheet()によって自動作成され，更に，サブルーチンprintBill()によって図2.4に示す請求書が印刷される。ここで，図2.3においてサブルーチンinputDate()及びサブルーチンcreateSumupSheet()は省略されている。また，手続き中のセル座標の指定は，行番号をR，列番号をCで表す。例えば，セルD3はR3C4（第3行，第4列）と表される。

```
Option Explicit
Dim bookname As String                    '文書名
Dim bill As String                        '請求書用紙の名
Dim sales As String                       'データベースの名
Dim sumup As String                       'ピボット表の名
Private lowend As Integer                 '下端の行数
Private rend As Integer                   '右端の列名
Private line As Integer                   '行
Private col As Integer                    '列
Private linetopaste As Integer            '請求書貼付け行
Private subtotal As Long                  '売上金額の小計
Private tax As Long                       '消費税

'請求書作成手続き
'-------------------------------------------------------------
Sub drawUpBill()
  bookname = ActiveWorkbook.Name          '文書名
  bill = "billform"                       '請求書用紙の名
  sales = "salesdata"                     'データベースの名
  sumup = "sumupsheet"                    'ピボット表の名
```

```
Sheets(bill).Select
   Range("G6").ClearContents                          '年月日を初期化
   Range("E11").ClearContents                         '社名を初期化
   Range("E18").ClearContents                         '請求額を初期化
   Range("C22:J31").ClearContents                     '明細を初期化
   Range("K32").Select                                'カーソルの退避
[ (1) ]                                               '日付入力用手続きの呼出し
[ (2) ]                                               'ピボット表の作成用手続きの呼出し
  Application.ScreenUpdating = True                   '画面の更新
[ (3) ]                                               '請求書の印刷用手続きの呼出し
End Sub
'  請求書の印刷手続き
'-------------------------------------------------------------------------------------
Private Sub printBill()
  lowend = Sheets(sumup).Range("A3").End(xlDown).Row                   '下端検出
  rend = Sheets(sumup).Range("B2").End(xlToRight). [ (4) ]             '右端検出
  Sheets(bill).Select
   For line = 3 To lowend - 1                                          'ピボット表の3行目～最下行-1
    Range("C22:J31"). [ (5) ]                                          '明細を初期化
    Cells(11, 5) = Sheets(sumup). [ (6) ] (line, 1)                    '顧客名
    linetopaste = 22                                                   '請求書用紙の金額の最上行
    For col = 2 To rend - 1
      If Sheets(sumup).Cells(line, col) <> "" Then
        Cells(linetopaste, 4) = Sheets(sumup).Cells(2, col)            '品名
        Cells(linetopaste, 7) = Sheets(sumup).Cells(line, col)         '売上
        linetopaste = [ (7) ] + 1
      End If
    Next
    [ (8) ] = Sheets(sumup).Cells(line, rend)                          '横計
    Cells(linetopaste, 6) = "小計"
    Cells(linetopaste, 7) = subtotal                                   '小計
    tax = Application. [ (9) ] (subtotal * 5 / 100, 0)
    Cells(linetopaste + 1, 6) = "消費税"
    Cells(linetopaste + 1, 7) = [ (10) ]                               '消費税
    Cells(linetopaste + 2, 6) = "合計"
    Cells(linetopaste + 2, 7) = subtotal + [ (10) ]                    '合計額
    Cells(18, 5) = subtotal + [ (10) ]                                 '請求額
    ActiveWindow.SelectedSheets.PrintPreview                           '印刷
   Next
 End Sub
```

図2.3　請求書作成のための手続き

このマクロプログラムを実行した結果の一部を図2.4に示す。



ファイル(F) 編集(E) 表示(V) 挿入(I) 書式(O) ツール(T) データ(D) ウィンドウ(W) ヘルプ(H) Acrobat(B)

請　　求　　書

平成14年7月14日

東京都港区芝公園3-5-22

東京国際（株）　　　　様　　　　（株）PCUA商事

Tel: (03)3438-1869
Fax: (03)5472-5338

下記のとおりご請求申し上げます。

ご請求額	209,790 円

月日	摘　要	金額	備考
	イメージスキャナ α	120,000	
	プリンタ Model 1	79,800	
	小計	199,800	
	消費税	9,990	
	合計	209,790	

salesdata　sumupsheet　billform

図2.4　マクロプログラムによって作成された請求書

解答群

01 linetopaste	02 tax	03 salesdata	04 billform
05 drawUpBill	06 inputDate	07 printBill	08 createSumupSheet
09 subtotal	10 Cells	11 Column	12 Row
13 Range	14 End	15 Round	16 ActiveWindow
17 ClearContents	18 Sub	19 Private Sub	20 Select

[第38回(2002年7月実施)パソコン認定試験３級より]

例題２の解説

表計算ソフトウェアを用いて業務応用プログラムの設計及び開発に関する問題です。この問題では売上表からピボット表を作成し，これらのデータから顧客別の請求書を作る手順を自動化（マクロ）するプログラミング言語に関する理解が求められています。

図2.3のマクロプログラムを上から順に見ていきます。まず，OptionExplicitで，プログラムで用いるすべての変数を宣言します。Dimは手続きで使用される変数の宣言で，別のサブルーチンの中でもその値を保持します。Privateはサブルーチンの手続きに関する局所変数です。

請求書作成の手続きはdrawUpBill()というメインルーチンで行われます。darwUpBill()では，文書・請求書用紙・データベース・ピボット表のファイル名をそれぞれの変数に格納して，請求書用紙の初期化を行ってから，日付入力・ピボット表の作成・請求書の印刷を行います。問題文からもわかるように，日付入力・ピボット表の作成・請求書の印刷に関しては，サブルーチンを呼び出して手続きを実行します。したがって，(1)には「06 inputDate」，(2)には「08 createSumupSheet」，(3)には「07 printBill」が入ります。これらは，右側に書いてあるコメント文を見れば明らかでしょう。

次に，請求書の印刷手続きprintBill()についてみていきます。(4)は上の行の下端検出の手続きを参考にして考えればよいでしょう。下端検出では，ピボット表のA3から1行ずつ下へ移動していき，下端まできたらその行の値をlowendに格納します。同様に考えて，右端検出ではB2から1列ずつ右へ移動し，右端まできたらその列の値をrendに格納すればよいので，(4)には「11 Column」が入ります。(5)はコメント文に初期化とあるので，drawUpBill()の6〜10行目の手続きを参考にすると，(5)には「17 ClearContents」が入ります。(6)は，ピボット表に記入されている顧客名を請求書用紙に出力する手続きです。顧客名はピボット表においてA列に記入されているので，各行のA列の値を請求書用紙の顧客名の欄（ここではE11）に格納すればよいことになります。したがって，(6)には「10 Cells」が入ります。

そして，printBill()の8行目から，請求書の詳細をピボット表から出力する手続きです。各々の顧客ごとにB列からF列までを順番に見ていき，該当のセルに金額が入力されていれば，その商品名と売上金額を請求書用紙に出力し，出力したら請求書用紙の出力先を1行下の行へと移動します。したがって，(7)には「01 linetopaste」が入ります。(8)は，ピボット表における横計，すなわち請求書の小計を表す変数が入ります。したがって，(8)は「09 subtotal」となります。(9)は，計算した小計に対する消費税を計算する手続きです。図2.4を見るとわかるように消費税は値を整数にまるめなければならないので，マクロプログラムに組み込まれている関数Roundを用います。したがって，(9)には「15 Round」が入ります。(10)は，“消費税”という文字列の右隣に出力していることや，合計額・請求額を求める計算式でsubtotalとの和をとっていることから，(10)には「02 tax」が入ります。

解説：パソコン認定試験研究会

■パーソナルコンピュータユーザ利用技術協会とは・・・

社団法人パーソナルコンピュータユーザ利用技術協会は，パソコンユーザの利用技術に関する調査，研究，普及啓発，情報の収集・提供および認定試験の実施，そして内外機関との交流，協力を行うことにより，情報リテラシの涵養を図り，わが国情報化社会の健全な発展に寄与することを目的に，平成3年5月に経済産業省認可の公益法人として設立された，全国を代表する唯一の機関です。

その一環として，パソコン利用技術向上のために，パーソナルコンピュータ利用技術認定試験（略称：パソコン認定試験）を全国規模で実施するなど，その教育的内容は高く評価されています。特に，パソコン認定試験合格者は，文部科学省が実施する大学入学資格検定の受検科目（情報関係基礎）が免除されます。

また，平成15年度から高等学校普通教科「情報」が新設されるにあたって，どのような知識・利用技術を身につけたらよいのか，利用者自身が主体的に情報を活用していく能力，すなわち，情報リテラシ（情報活用能力）の涵養を図るとともに，小中高校生の育成と情報化社会の健全な発展に寄与することを目指しています。

ホームページ　http://www.pcua.or.jp/

協力：（株）富士通南九州システムエンジニアリング

はじめよう！ Microsoft Officeで BASICプログラミング No.10

第10回 コンピュータ対戦型ポーカーに挑戦(1)

●文/椛島たけし

今月から対戦型ゲームの作成を2～3回に分けて行ないます。題材として「コンピュータ対戦型ポーカー」の作成を目標にします。今月は，ゲームのルールをプログラミングします。プログラムの作成はMicrosoft Excel/Wordのどちらでも OKです。

対戦型ポーカーのゲーム性を考える

2002年2月号～4月号で「コンピュータ対戦型オセロ」を作成しました。このときは，Excelのセルを盤面に見立てていました。また，「手の先を読む」ことを中心に，手の先読みの理論を紹介しました。

ポーカーの場合，相手の手は見えませんし，交換した(捨てた)カードも伏せておきますので相手の手を読む術はあまりありません。本当のポーカーであれば，表情で読むのですが……。コンピュータの場合は，本当にポーカーフェースですね。とりあえず，相手の手を読むことは行なわずに，ゲームとして成立することを考えてみましょう。

ゲームのルール

一般的なポーカー(ドローポーカー)のルールは次のようになっています

手順1.各プレイヤーに5枚のカードが配られる

手順2.プレイヤーは交換するカードを選ぶ(0枚～5枚)

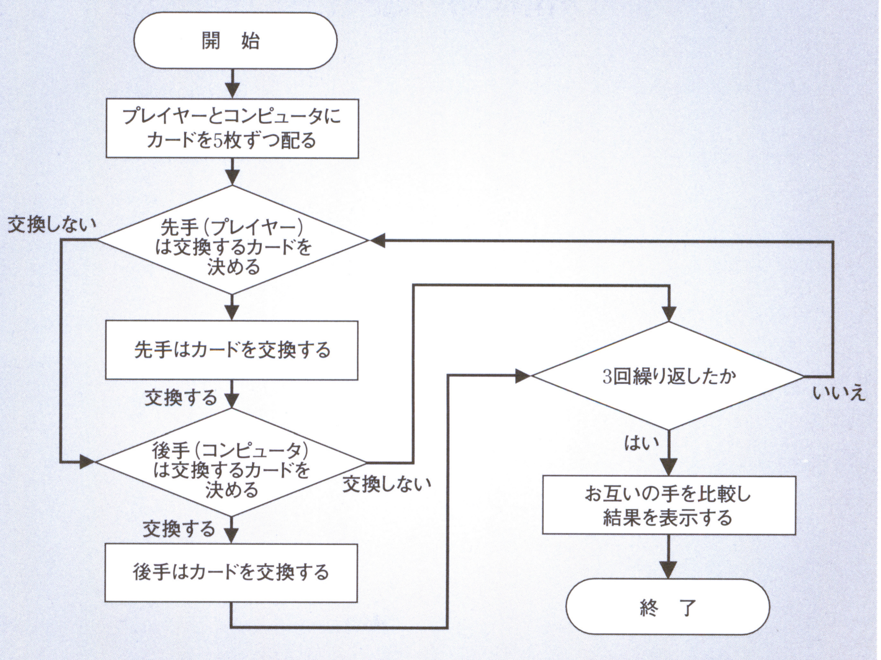
手順3.カードを交換しない場合は，手順8まで中断する

手順4.プレイヤーは交換するカードを伏せて捨てる

手順5.プレイヤーは捨てたカードと同じ枚数のカードを引く

《第1表》 ポーカーの役と強さ

	役の名前	内容
↑強い	ロイヤルストレートフラッシュ	"10,11,12,13,Aのストレートでフラッシュ"
	ストレートフラッシュ	A～9ではじまるストレートでフラッシュ
	フォーカード	同じ数字が4枚
	フルハウス	スリーカードとワンペアの組み合わせ
	フラッシュ	5枚がすべて同じ柄
	ストレート	5枚の数字が連続している
	スリーカード	同じ数字が3枚
	ツーペア	ペアが2組
↓弱い	ワンペア	同じ数字が2枚(ペア)が1組



▲《第1図》 ポーカーのルールの流れ

《第2表》ポーカーの役と交換しない場合の確率

役	手の数	確率
ロイヤルストレートフラッシュ	4	0.00015%
ストレートフラッシュ	36	0.00139%
フォーカード	624	0.02401%
フルハウス	"3,744 "	0.14406%
フラッシュ	"5,108 "	0.19654%
ストレート	"10,200 "	0.39246%
スリーカード	"54,912 "	2.11285%
ツーペア	"123,552 "	4.75390%
ワンペア	"1,098,240 "	42.25690%
役無し	"1,302,540 "	50.11774%
合計	"2,598,960 "	100.00000%

手順6.手順2〜5を他のプレイヤーが行なう

手順7.手順2〜手順6を3回繰り返す

手順8.各プレイヤーは5枚のカードを開き，勝ち負けを決める

コンピュータとプレイヤーの2人の対戦を想定し，これらのルールを元に流れ図を作成してみましょう(**第1図**)。

コンピューターが手を決める理論

さて，ポーカーで手を決める(交換する)ためには，状況から狙う手を決める必要があります。普通，状況には次のようなものがあります。

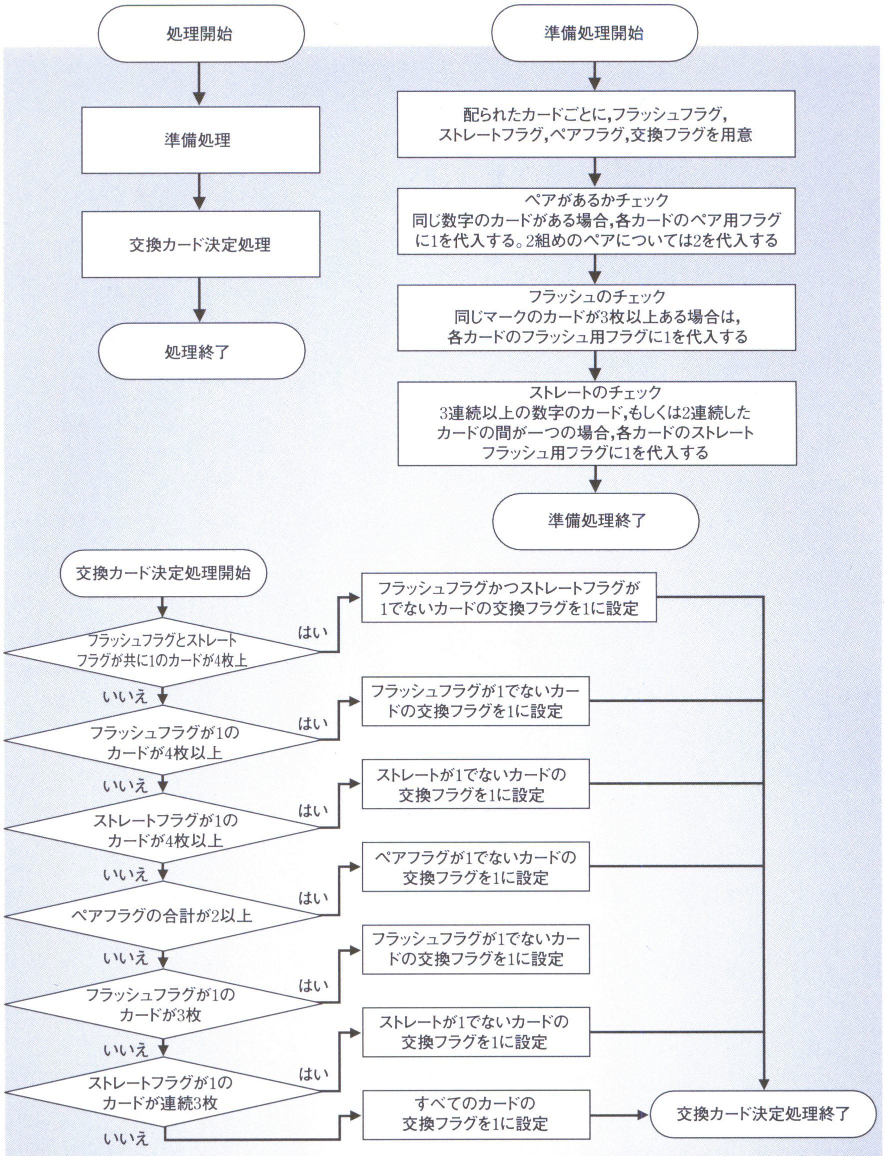
①配られたカードの組み合わせ

②勝ち負けの状態

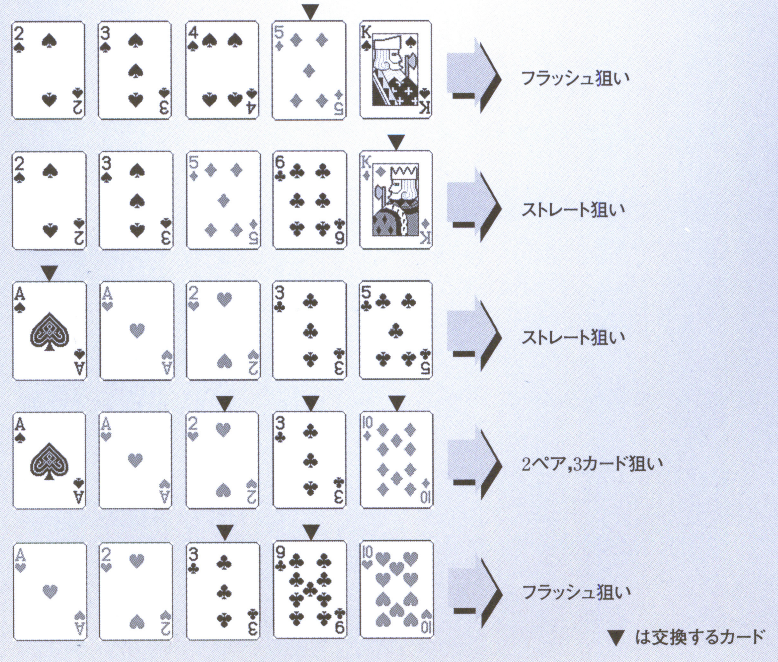
③相手の状況(表情など)

まずは，②，③については置いておいて，まずは配られたカードから狙う手を決める方法を考えてみます。

ポーカーの役は，**第1表**のようになっています。当然，強い役ほど作るのが難しくなります。もちろん，なるべく高い手を確実に作りたいですね。まずは，筆者がポーカーをするときの役づくりの思考をなるべく単純化して，次のような流れ図を考えてみました(**第2図**)。ストレートやフラッシュの可能性があるカードは4枚以上の場合は狙います。3枚以下の場合は，ペア系の手を優先させます。ペアがない場合，ストレートやフラッシュの可能性があるカードが3枚あると，ストレートやフラッシュを狙います(**第3図**)。できるだけ確率が高いと思われる選択をしていますが本



▲《第2図》コンピュータが交換するカードを決める流れ図



◀《第3図》流れ図の動きの具体例

当にこれで良いのか疑問が残るところです(笑い)。皆さんも，自分なりの考えを流れ図に反映させてみてください。

いよいよプログラミング

さて，ルールやコンピュータが手をきめる理論ができたところでプログラミングです。今回はWordを使用していますが，ExcelでもOKです。リストに一箇所だけ違いがありますので，コメントに注意しておいてください。**第4図**を参考に画面を作成しましょう。また，あらかじめトランプを表示するための画像ファイルを作成しておく必要があります(ペーマガのホームページからダウンロードしてもOKです)。**リスト**についてはちょっと長くなっていますが，流れ図と比較してたどってみてください。また，プログラミングの具体的な方法については**第X図**を参照してください。

実行結果は?

実行例は**第5図**です。何度か対戦してみると，それなりにコンピュータも健闘しているのではないでしょうか？ ただし，役によって点数をつけるなどの処理を組み込んでいませんので，単純に勝敗だけではプレイヤーに分があるかもしれません。次回は，より確率論を反映させて，コンピュータをより強くすることにしましょう。

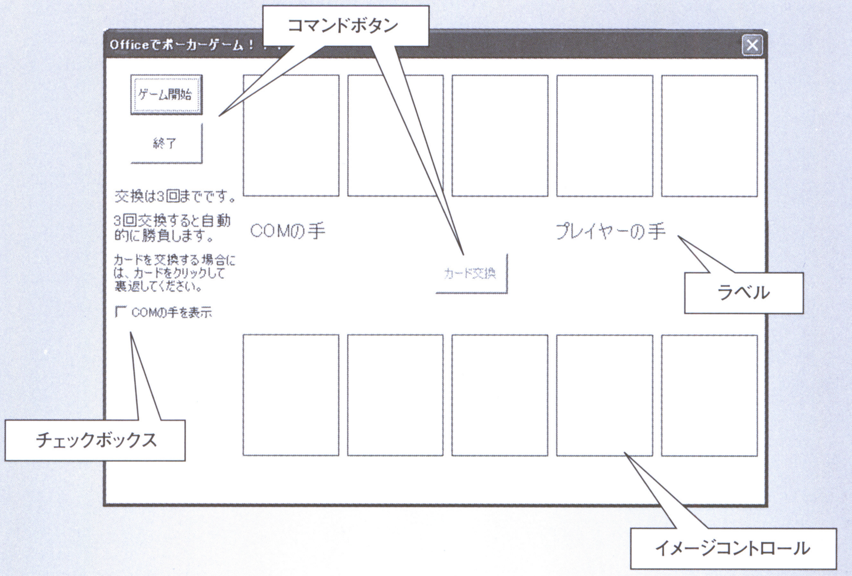
Technical Column

ポーカーの役の難易度を計算する

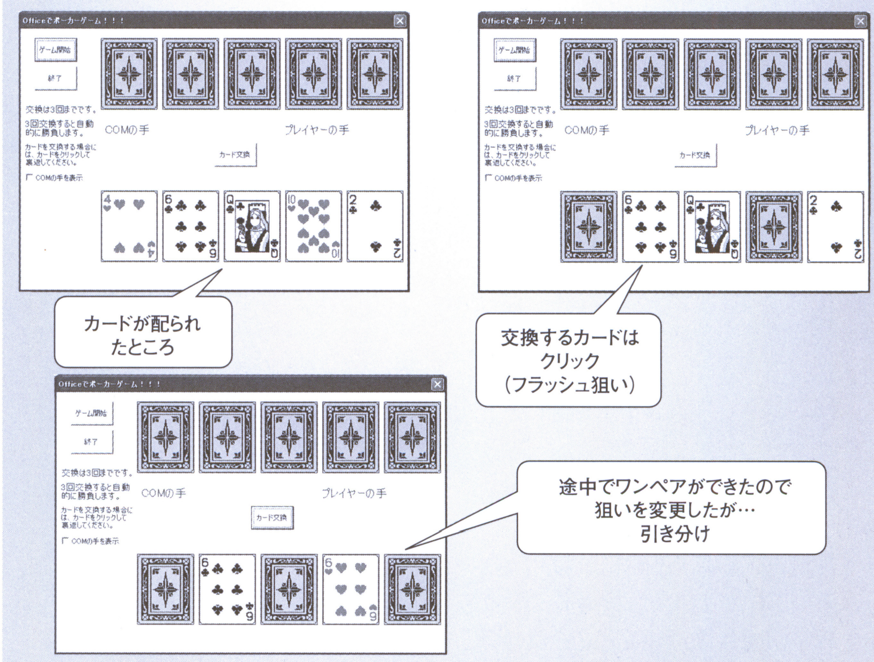
たとえば，よく迷うストレートとフラッシュが最初に配られた5枚で完成する確率を計算してみます。

52枚のカードから5枚のカードが配られた手のパターンは次の式で求められます。
(52×51×50×49×48)÷(5×4×3×2×1) = 2598960通りです。

ストレートはAから9の間ではじまる連続した数ですが，計算すると
(4×4×4×4×4)×10 = 10240 通りと



▲《第4図》ポーカーゲームの画面



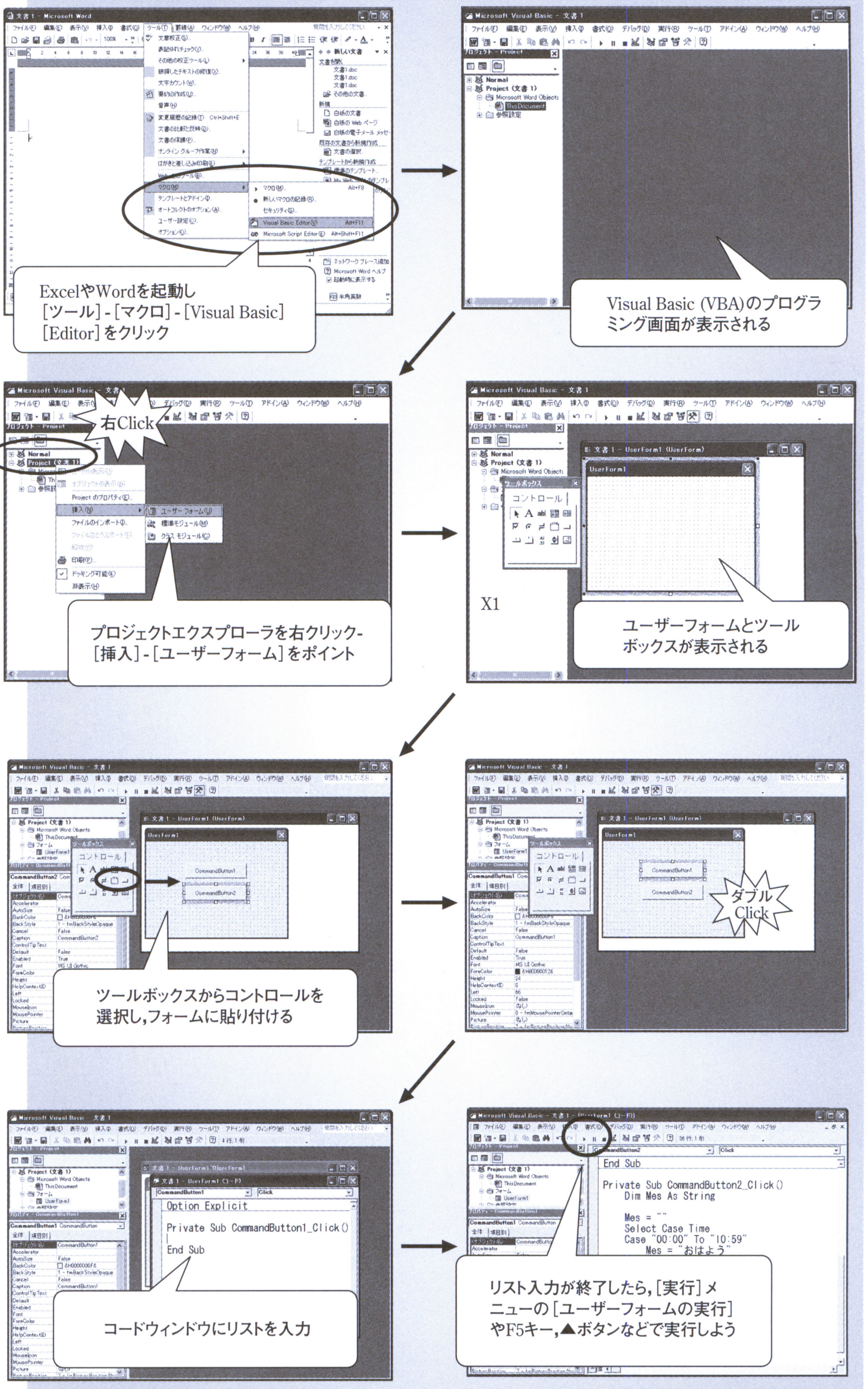
▲《第5図》プログラム実行例。とりあえず実行できるようになった

なります(※注)。
　したがって確率は，10240÷2598960≒0.0039・・・で約0.4%ですね。

フラッシュは
(13×12×11×10×9)÷(5×4×3×2×1)×4＝5148通りとなります(※注)。したがって確率は，5148÷2598960≒0.0019・・・で約0.2%ですね。
※注 ストレートフラッシュ含む

　このように，ポーカーの手の強さは数学的にも正しい？わけです。
　第2表が役とその確率となります。何も役がない確率は約50%となります。ただし，手が進むと条件が一変します。
　一巡目に，続きのカードが3枚ある場合と，同じ図柄が3枚ある場合に，ストレートとフラッシュができる確率はどのようにちがうのでしょうか？
　♠3，♥4，♥5，♣9，♥Jという手の場合，ストレート狙いかフラッシュ狙いか悩むところです。
・ストレート狙いの確率
　ストレートを狙う場合，♣9と♥Jを捨てます。交換後の1枚目にAが来て，2枚目に2が来る確率が，4／47×4／46。このパターンで計算すると，連続した3枚からストレートが完成する確率は(16×6)／(47×46)となります。4.4%程度ですね。
・フラッシュ狙いの確率
フラッシュを狙う場合，♠3と♣9を捨て，♥狙いです。同じ図柄3枚から2枚交換でフラッシュが完成する確率は，♥の残り枚数を考慮して，(10／47)×(9／46)となります。約4.1%です。
　フラッシュがストレートより強いことを考えると，状況によってはフラッシュでもOKですね。また，♠2，♥3，♥4，♣9，♥Jという組み合わせの場合は，ストレートが完成する確率が約3%となり，先ほどのケースより分が悪くなります。この場合は，フラッシュ狙いが正解ということになります。

　今回のプログラムでは，狙う役を決めるアルゴリズムは筆者の好みでしたが，次回は理論的に考えてみる必要がありそうです。



▲《第X図》プログラム作成と実行

《リスト1》「ポーカーゲーム」のプログラム・リスト

```
Option Explicit

'手役データの設定
Private Enum typRole
    ROLE_LOYAL_STRAIGHT_FLASH = 256
    ROLE_STRAIGHT_FLASH = 128
    ROLE_FOUR_CARDS = 64
    ROLE_FULL_HOUSE = 24
    ROLE_FLASH = 16
    ROLE_STRAIGHT = 8
    ROLE_THREE_CARDS = 4
    ROLE_TWO_PAIR = 2
    ROLE_ONE_PAIR = 1
    NO_ROLE = 0
End Enum

'カード定義型
Private Type typCard
    CardMark As String  'カードのマーク
    CardPoint As Long    'カードの数字
    ImageParh As String   'カードの画像パス
End Type

'手札定義型
Private Type typHandCard
    udtHand As typCard   'カードデータ
    StraightFlg As Long       'ストレートフラグ
    FlashFlg As Long       'フラッシュフラグ
    PairFlg As Long       'ペアフラグ
    ChangeFlg As Long   '交換フラグ
End Type

Const HANDS_MAX = 5  '手札の枚数
Const CARDS_MAX = 52 'カードの枚数

Private mudtCards(1 To CARDS_MAX) As typCard   '全データを含むカード配
列
Private mudtShuffles(1 To CARDS_MAX) As typCard '52枚をシャッフルしたカ
ード配列
Private mudtCOM(1 To HANDS_MAX) As typHandCard   'COMの手札
Private mudtUser(1 To HANDS_MAX) As typHandCard  'プレイヤーの手札
Private mudtBack As typCard                'カードの裏側データ
Private mlngMax As Long                     'カードの最大枚数
Private mlngChangeCt As Long                'カード交換回数カウント
Private mlngDispFlg As Long                 'カード表示チェック


'COMの手を表示させるかどうかのチェック
Private Sub CheckBox1_Click()
    'チェックが入っている場合
    If CheckBox1.Value = True Then
        mlngDispFlg = 1 '表示チェックフラグを1にして表示
    Else
        mlngDispFlg = 0 '表示チェックフラグを0にして表示しない
    End If
End Sub

'ゲーム開始ボタン
Private Sub CommandButton1_Click()
    Dim udtCOMRole As typRole  'COMの手役変数
    Dim udtUserRole As typRole 'プレイヤーの手役変数

    '初期化
    mlngChangeCt = 1
    mlngDispFlg = 0
    Label3.Caption = ""
    Label4.Caption = ""

    'COMの手を表示するか判断
    If CheckBox1.Value = True Then
        mlngDispFlg = 1
    End If

    CommandButton2.Enabled = True 'ボタンを使用可能にする

    Call procStartGame 'カード準備

    'カードチェック
    Call procPairCheck(mudtCOM)      'ペアチェック
    Call procFlashCheck(mudtCOM)     'フラッシュチェック
    Call procStraightCheck(mudtCOM)  'ストレートチェック
    Call procPairCheck(mudtUser)     'ペアチェック
    Call procFlashCheck(mudtUser)    'フラッシュチェック
    Call procStraightCheck(mudtUser) 'ストレートチェック

    Call procRoleChoice(mudtCOM) 'COMの手役選択を実行

    '手役の決定
    udtCOMRole = procRoleCheck(mudtCOM)
    udtUserRole = procRoleCheck(mudtUser)

    CommandButton2.SetFocus 'カード交換ボタンにフォーカスをあてる

    Me.Repaint 'UserFormの再描画
End Sub

'カード交換ボタン
Private Sub CommandButton2_Click()
    Dim udtCOMRole As typRole  'COMの手役変数
    Dim udtUserRole As typRole 'プレイヤーの手役変数
    Dim lngCt As Long           'ループカウント

    Call procRoleChoice(mudtCOM) 'COMの手役選択を実行

    'カード交換
    Call procCardChange(mudtCOM)
    Call procCardChange(mudtUser)

    Call procDisplay 'カードイメージの表示

    'カードチェック
    Call procPairCheck(mudtCOM)      'ペアチェック
    Call procFlashCheck(mudtCOM)     'フラッシュチェック
    Call procStraightCheck(mudtCOM)  'ストレートチェック
    Call procPairCheck(mudtUser)     'ペアチェック
    Call procFlashCheck(mudtUser)    'フラッシュチェック
    Call procStraightCheck(mudtUser) 'ストレートチェック

    '手役の決定
    udtCOMRole = procRoleCheck(mudtCOM)
    udtUserRole = procRoleCheck(mudtUser)

    'カード交換ボタンを押した回数の判断
    mlngChangeCt = mlngChangeCt + 1 '交換回数のカウントを1増やす
    If mlngChangeCt > 3 Then    '3回を超えた場合
        Call procMatch '勝ち負け判定用プロシージャの呼び出し
    End If

End Sub

'終了ボタン
Private Sub CommandButton3_Click()
    Unload Me
End Sub

'Image10をクリックしたときの動作
Private Sub Image10_Click()
    'ゲーム開始前、もしくは勝負後はクリックできない
    If mlngChangeCt > 0 And mlngChangeCt <= 3 Then
        Call procImageClick(mudtUser(5), Image10)
    End If
End Sub

'Image6をクリックしたときの動作
Private Sub Image6_Click()
    'ゲーム開始前、もしくは勝負後はクリックできない
    If mlngChangeCt > 0 And mlngChangeCt <= 3 Then
        Call procImageClick(mudtUser(1), Image6)
    End If
End Sub

'Image7をクリックしたときの動作
Private Sub Image7_Click()
    'ゲーム開始前、もしくは勝負後はクリックできない
    If mlngChangeCt > 0 And mlngChangeCt <= 3 Then
        Call procImageClick(mudtUser(2), Image7)
    End If
End Sub

'Image8をクリックしたときの動作
Private Sub Image8_Click()
    'ゲーム開始前、もしくは勝負後はクリックできない
```

```
    If mlngChangeCt > 0 And mlngChangeCt <= 3 Then
        Call procImageClick(mudtUser(3), Image8)
    End If
End Sub

'Image9をクリックしたときの動作
Private Sub Image9_Click()
    'ゲーム開始前、もしくは勝負後はクリックできない
    If mlngChangeCt > 0 And mlngChangeCt <= 3 Then
        Call procImageClick(mudtUser(4), Image9)
    End If
End Sub

'ユーザフォームを表示
Private Sub UserForm_Initialize()
    Dim strParh As String      'Wordドキュメントのパス
    Dim strImagePath As String '画像ファイルのフォルダパス
    Dim strIni As String       '設定ファイルのパス
    Dim lngFreeFile As Long    'ファイル変数
    Dim lngCt As Long          'ループカウント
    Dim strMark As String      'マーク変数
    Dim lngPoint As Long       '数字変数
    Dim strInput As String     'ファイルから読み込んだデータ
    Dim strTitle As String     'タイトル保存変数
    Dim strFolder As String    'フォルダパス保存変数

    mlngChangeCt = 0 '交換回数の初期化

    strParh = ThisDocument.Path & "¥"   'Wordドキュメントのパスを設定
    'Excelシート上のフォームで利用する場合のExcelシートのパスの取得方法は
以下の通りになります。
    'strParh = ThisWorkbook.Path & "¥"   'Excelシートのパスを設定
    strIni = strParh & "Game.ini"       'ゲーム設定ファイルのパスを設定

    '各設定の読み込み
    lngFreeFile = FreeFile
    Open strIni For Input As #lngFreeFile '設定ファイルの読み込み
    Do Until EOF(lngFreeFile) 'ファイルが終了するまで読み込む
        Line Input #lngFreeFile, strInput
        '読み込んだ行が持つデータによって、利用データの判別
        Select Case InStr(strInput, "=")
        Case Is = 6  'タイトル表示
            strTitle = Mid$(strInput, 7)
            UserForm1.Caption = strTitle
        Case Is = 10 'カード裏面画像ファイル名
            mudtBack.ImageParh = Mid$(strInput, 11)
        Case Is = 12 '画像ファイルの保存先フォルダ
            strFolder = Mid$(strInput, 13)
            strImagePath = strParh & strFolder
        End Select
    Loop
    Close #lngFreeFile '設定ファイルの終了

    'カードデータの読み込み
    lngCt = 1
    strIni = strParh & "Cards.ini" 'カードデータ設定ファイルを読み込む
    Open strIni For Input As #lngFreeFile
    Do Until EOF(lngFreeFile)
        Input #lngFreeFile, strMark, lngPoint, strParh
        mudtCards(lngCt).CardMark = strMark
        mudtCards(lngCt).CardPoint = lngPoint
        mudtCards(lngCt).ImageParh = strImagePath & strParh
        lngCt = lngCt + 1
    Loop
    Close #lngFreeFile

    'カード裏面用パスの設定
    mudtBack.ImageParh = strImagePath & mudtBack.ImageParh
End Sub

'ゲーム開始の準備を行うプロシージャ - カードのシャッフル及び表示
Private Sub procStartGame()
    Dim udtCopy As typCard 'シャッフル時の保存変数
    Dim lngCt As Long        'ループカウント変数
    Dim lngCOMCt As Long     'COMにシャッフルした何枚目のカードを配るかカ
ウントする変数
    Dim lngUserCt As Long    'ユーザにシャッフルした何枚目のカードを配るか
カウントする変数
    Dim lngChangeCt As Long '交換枚数カウント
    Dim lngRnd As Long       'ランダム変数

    mlngMax = CARDS_MAX 'カードの最大枚数の設定
    For lngCt = 1 To mlngMax 'カードデータを複製
        mudtShuffles(lngCt) = mudtCards(lngCt)
    Next lngCt

    'カードシャッフルの実行
    For lngCt = 1 To mlngMax
        Randomize    ' 乱数発生ルーチンを初期化します。
        lngRnd = Int(Rnd * (mlngMax - 0) + 1)
        udtCopy = mudtShuffles(lngCt)
        mudtShuffles(lngCt) = mudtShuffles(lngRnd)
        mudtShuffles(lngRnd) = udtCopy
    Next lngCt

    'カードを配る
    For lngCt = 1 To HANDS_MAX
        lngUserCt = lngCt * 2
        lngCOMCt = lngUserCt - 1
        mudtCOM(lngCt).udtHand = mudtShuffles(lngCOMCt)
        mudtCOM(lngCt).ChangeFlg = 0
        mudtUser(lngCt).udtHand = mudtShuffles(lngUserCt)
        mudtUser(lngCt).ChangeFlg = 0
        lngChangeCt = lngChangeCt + 2 '交換枚数のカウント
        mlngMax = mlngMax - 2 '最大枚数から2枚減らす
    Next lngCt

    For lngCt = 1 To mlngMax '使用した枚数分前にずらす
        mudtShuffles(lngCt) = mudtShuffles(lngCt + lngChangeCt)
    Next lngCt

    Call procDisplay 'カードの表示
End Sub

'カード交換プロシージャ
Private Sub procCardChange(Hands() As typHandCard)
    Dim lngCt As Long        'ループカウント
    Dim lngChangeCt As Long '交換枚数カウント

    For lngCt = 1 To HANDS_MAX 'カード交換の実行
        If Hands(lngCt).ChangeFlg = 1 Then
            lngChangeCt = lngChangeCt + 1
            Hands(lngCt).udtHand = mudtShuffles(lngChangeCt)
        End If
        Hands(lngCt).PairFlg = 0     'ペアフラグの初期化
        Hands(lngCt).FlashFlg = 0    'フラッシュフラグの初期化
        Hands(lngCt).StraightFlg = 0 'ストレートフラグの初期化
        Hands(lngCt).ChangeFlg = 0   'カード交換フラグの初期化
    Next lngCt

    mlngMax = mlngMax - lngChangeCt '最大枚数から交換を実行した枚数を減
らす
    For lngCt = 1 To mlngMax        'カード配列のデータを前にずらす
        mudtShuffles(lngCt) = mudtShuffles(lngCt + lngChangeCt)
    Next lngCt

End Sub

'ペアチェック
Private Sub procPairCheck(Hands() As typHandCard)
    Dim lngCt As Long          'ループカウント
    Dim lngIndex1 As Long     '1枚目を表すカウント
    Dim lngIndex2 As Long     '2枚目を表すカウント
    Dim lngPairPiece As Long 'ペアの個数をカウント
    Dim lngPairFlg As Long   'ペアが成立しているか確認するフラグ

    For lngCt = 1 To HANDS_MAX '手札からフラッシュフラグを初期化
        Hands(lngCt).PairFlg = 0 '初期化
    Next lngCt

    For lngIndex1 = 1 To HANDS_MAX - 1 '2枚の数値を比較して同じ値のもの
をチェック
        lngPairFlg = 0 'ペア成立フラグの初期化
        '既にペア確認済みのカード以外をチェック
        If Hands(lngIndex1).PairFlg = 0 Then
            lngPairPiece = lngPairPiece + 1 'ペアの個数カウントを1増や
す
            For lngIndex2 = lngIndex1 + 1 To HANDS_MAX
                If Hands(lngIndex1).udtHand.CardPoint =
Hands(lngIndex2).udtHand.CardPoint And Hands(lngIndex2).PairFlg = 0
Then
                    Hands(lngIndex1).PairFlg = lngPairPiece 'ペアの個数
をペアフラグに代入
```

```
                Hands(lngIndex2).PairFlg = lngPairPiece 'ペアの個数
をペアフラグに代入
                lngPairFlg = 1 'ペア成立フラグを1にする
            End If
        Next lngIndex2
        If lngPairFlg = 0 Then 'ペアがなかった場合
            lngPairPiece = lngPairPiece - 1 'ペアの個数カウントを1
減らす
        End If
    End If
    Next lngIndex1
End Sub

'フラッシュチェック
Private Sub procFlashCheck(Hands() As typHandCard)
    Dim lngCt As Long           'ループカウント
    Dim lngSPADE As Long        'スペード枚数カウント
    Dim lngHEART As Long        'ハート枚数カウント
    Dim lngDIAMOND As Long      'ダイヤ枚数カウント
    Dim lngCLUB As Long         'クラブ枚数カウント
    Dim strMarkCheck As String 'マーク名保存変数

    For lngCt = 1 To HANDS_MAX '手札からフラッシュフラグを初期化
        Hands(lngCt).FlashFlg = 0 '初期化
    Next lngCt

    For lngCt = 1 To HANDS_MAX '各マークの数をチェック
        Select Case Hands(lngCt).udtHand.CardMark
        Case Is = "スペード"
            lngSPADE = lngSPADE + 1
        Case Is = "ハート"
            lngHEART = lngHEART + 1
        Case Is = "ダイヤ"
            lngDIAMOND = lngDIAMOND + 1
        Case Is = "クラブ"
            lngCLUB = lngCLUB + 1
        End Select
    Next lngCt

    Select Case 3 '3枚以上のマークをチェック
    Case Is <= lngSPADE
        strMarkCheck = "スペード"
    Case Is <= lngHEART
        strMarkCheck = "ハート"
    Case Is <= lngDIAMOND
        strMarkCheck = "ダイヤ"
    Case Is <= lngCLUB
        strMarkCheck = "クラブ"
    End Select

    For lngCt = 1 To 5 '3枚以上あるマークを持つ手札をチェック
        If Hands(lngCt).udtHand.CardMark = strMarkCheck Then
            Hands(lngCt).FlashFlg = 1 'フラッシュチェックフラグを1にす
る
        Else
            Hands(lngCt).FlashFlg = 0 'フラッシュチェックフラグを0にす
る
        End If
    Next lngCt
End Sub

'ストレートチェック
Private Sub procStraightCheck(Hands() As typHandCard)
    Dim udtWork(5) As typHandCard '手札編集用配列
    Dim udtPreserve As typHandCard 'ソート時の手札保存用変数
    Dim lngMax As Long            '最大枚数を保存する変数
    Dim lngDifSum As Long         '1番目と2番目の差と2番目と3番目の差
の合計
    Dim lngCt As Long             'ループカウント
    Dim lngCtMAX As Long          'ループカウントの最大値
    Dim lngSlideLoop As Long      '配列をずらすループカウント
    Dim lngSlideCheck As Long     '配列をずらしたかチェックフラグ
    Dim lngCompareLoop As Long    '値を比較するループカウント
    Dim lngIndex As Long          '配列インデックス
    Dim lngIndex1 As Long         '1番目の配列インデックス
    Dim lngIndex2 As Long         '2番目の配列インデックス
    Dim lngIndex3 As Long         '3番目の配列インデックス
    Dim lngIndex4 As Long         '4番目の配列インデックス
    Dim lngIndex5 As Long         '5番目の配列インデックス
    Dim lngDif1to2 As Long        '1番目と2番目の数字の差を保存する変
数
    Dim lngDif2to3 As Long        '2番目と3番目の数字の差を保存する変
数
    Dim lngDif3to4 As Long        '3番目と4番目の数字の差を保存する変数
    Dim lngDif4to5 As Long        '4番目と5番目の数字の差を保存する変数
    Dim lngStraightFlg As Long    '4枚連続で並んでいるかのチェックフラ
グ

    lngMax = HANDS_MAX '並びをチェックする最大枚数

    '手札のコピー
    For lngCt = 1 To lngMax
        Hands(lngCt).StraightFlg = 0  'ストレートフラグの初期化
        udtWork(lngCt) = Hands(lngCt) '手札のコピー
    Next lngCt
    '手札の数字を利用して逆順に並び替える
    For lngCt = 1 To HANDS_MAX - 1 '1番目と2番目を比較して、2番目が大き
ければ入れ替える
        For lngCompareLoop = lngCt + 1 To HANDS_MAX
            If udtWork(lngCt).udtHand.CardPoint <
udtWork(lngCompareLoop).udtHand.CardPoint Then
                udtPreserve = udtWork(lngCt)               '1番目のデータ
を保存
                udtWork(lngCt) = udtWork(lngCompareLoop) '2番目のデータ
を1番目の配列の場所に保存
                udtWork(lngCompareLoop) = udtPreserve    '保存した1番目
のデータを2番目の配列の場所に保存
            End If
        Next lngCompareLoop
    Next lngCt

    '同じ数字が複数ある場合
    For lngCt = 1 To HANDS_MAX - 1 '1枚目から4枚目までループ
        lngSlideCheck = 0 '配列をずらしたかのフラグを0にする
        For lngCompareLoop = lngCt + 1 To lngMax '最大枚数までループ
            '同じ数字のカードがあったら後ろにずらす
            'フラッシュチェックが入っているものを優先する
            If udtWork(lngCt).udtHand.CardPoint =
udtWork(lngCompareLoop).udtHand.CardPoint And
udtWork(lngCompareLoop).udtHand.CardPoint <> 0 Then
                If udtWork(lngCompareLoop).FlashFlg = 0 Then 'フラッシュ
フラグがない場合
                    lngIndex = lngCt + 1          '2番目のインデックス
に後ろからずらす
                Else
                    lngIndex = lngCt              '1番目のインデックス
に後ろからずらす
                End If
                For lngSlideLoop = lngIndex To lngMax - 1
                    udtWork(lngSlideLoop) = udtWork(lngSlideLoop + 1) '
カードを前にずらす
                Next lngSlideLoop
                udtWork(lngMax).udtHand.CardPoint = 0 '一番後ろのカード
の値を0にする
                lngMax = lngMax - 1
            End If
        Next lngCompareLoop
    Next lngCt

    '3枚の組み合わせからストレートができるか判断
    For lngCt = 1 To lngMax
        lngStraightFlg = 0 'ストレートチェックフラグの初期化
        '1番目と2番目の数字の差を求める
        lngIndex1 = lngCt                     '1番目の配列インデックスの
設定
        lngIndex2 = lngIndex1 Mod lngMax + 1 '2番目の配列インデックスの
設定
        lngDif1to2 = udtWork(lngIndex1).udtHand.CardPoint -
udtWork(lngIndex2).udtHand.CardPoint
        If lngDif1to2 = -12 Then lngDif1to2 = 1  '1番目と2番目の差が-12
のとき
        If lngDif1to2 < 0 Then lngDif1to2 = 3    '1番目と2番目の差が0よ
り小さいとき
        '2番目と3番目の数字の差を求める
        lngIndex3 = lngIndex2 Mod lngMax + 1 '3番目の配列インデックスの
設定
        lngDif2to3 = udtWork(lngIndex2).udtHand.CardPoint -
udtWork(lngIndex3).udtHand.CardPoint
        If lngDif2to3 = -12 Then lngDif2to3 = 1  '2番目と3番目の差が-12
のとき
        If lngDif2to3 < 0 Then lngDif2to3 = 3    '2番目と3番目の差が0よ
り小さいとき
        '求めた差を合計
        lngDifSum = lngDif1to2 + lngDif2to3
```

```
        '合計の値からストレートになるか判断
        Select Case lngDifSum
        Case 2, 3
            '3つの数字で最大の数字が3より大きい場合
            If udtWork(lngIndex1).udtHand.CardPoint > 3 Then
                udtWork(lngIndex1).StraightFlg = 1
                udtWork(lngIndex2).StraightFlg = 1
                udtWork(lngIndex3).StraightFlg = 1
            '3枚の最大の数字が3である場合
            ElseIf udtWork(lngIndex1).udtHand.CardPoint = 3 Then
                udtWork(lngIndex1).StraightFlg = 1
                udtWork(lngIndex2).StraightFlg = 1
                If lngDifSum = 2 Then '3枚連続の場合、3番目のストレート
フラグを1にする
                    udtWork(lngIndex3).StraightFlg = 1
                End If
                Exit For
            '3枚の最大の数字が2の場合
            ElseIf udtWork(lngIndex1).udtHand.CardPoint = 2 Then
                udtWork(lngIndex1).StraightFlg = 1
                If lngDif1to2 = 1 Then '1番目と2番目が連続している場合
                    udtWork(lngIndex2).StraightFlg = 1
                End If
                Exit For
            End If
            '3番目と4番目の差を求める
            lngIndex4 = lngIndex3 Mod lngMax + 1 '4番目の配列インデック
スの設定
            lngDif3to4 = udtWork(lngIndex3).udtHand.CardPoint -
udtWork(lngIndex4).udtHand.CardPoint
            If lngDif3to4 = -12 And
udtWork(lngIndex4).udtHand.CardPoint = 1 And udtWork(1).StraightFlg = 1
Then lngDif3to4 = 1
            If lngDif2to3 < 0 Then lngDif2to3 = 3    '3番目と4番目の差
が0より小さいとき
            Select Case lngDif3to4 '求めた差から4番目にチェックを入れる
か判断
            Case Is = 1 '差が1の場合
                udtWork(lngIndex4).StraightFlg = 1
                If lngDifSum = 2 Then
                    lngStraightFlg = 1 '4枚が連続した数字の場合、フラグ
を1にする
                Else
                    lngStraightFlg = 0 'それ以外はフラグを0にする
                End If
            Case Is = 2 '差が2の場合
                If lngDifSum = 2 Then
                    udtWork(lngIndex4).StraightFlg = 1
                    lngStraightFlg = 0
                End If
            Case Else 'それ以外の場合
                lngStraightFlg = 0
            End Select
            Exit For
        End Select
    Next lngCt

    If lngStraightFlg = 1 And lngMax = HANDS_MAX Then '最大枚数が5枚で、
4枚が連続した数字の場合
        '4番目と5番目の数字の差を求めて、差が1の場合はストレートチェッ
クフラグを1にする
        lngIndex5 = lngIndex4 Mod lngMax + 1 '4番目の配列インデックスの
設定
        lngDif4to5 = udtWork(lngIndex4).udtHand.CardPoint -
udtWork(lngIndex5).udtHand.CardPoint
        If lngDif4to5 = 1 Then
            udtWork(lngIndex5).StraightFlg = 1
        End If
    End If

    '元の配列のカードデータと一致したコピーのカードデータを代入
    For lngCt = 1 To HANDS_MAX
        For lngCompareLoop = 1 To lngMax
            If Hands(lngCt).udtHand.CardPoint =
udtWork(lngCompareLoop).udtHand.CardPoint And _
                            Hands(lngCt).udtHand.CardMark =
udtWork(lngCompareLoop).udtHand.CardMark Then
                Hands(lngCt) = udtWork(lngCompareLoop)
            End If
        Next lngCompareLoop
    Next lngCt
End Sub
```

```
'手役をチェックするプロシージャ
Private Function procRoleCheck(Hands() As typHandCard) As typRole
    Dim lngCt As Long                 'ループカウント
    Dim lngPairCt As Long             'ペアフラグの個数カウント
    Dim lngFlashCt As Long            'フラッシュフラグの個数カウント
    Dim lngStraightCt As Long         'ストレートフラグの個数カウント
    Dim lngStraightFlashCt As Long 'ストレートとフラッシュにチェックがあ
る個数カウント

    'カードの各フラグの確認
    For lngCt = 1 To HANDS_MAX
        lngPairCt = Hands(lngCt).PairFlg + lngPairCt
        lngFlashCt = Hands(lngCt).FlashFlg + lngFlashCt
        lngStraightCt = Hands(lngCt).StraightFlg + lngStraightCt
        If (Hands(lngCt).FlashFlg And Hands(lngCt).StraightFlg) = 1 Then
            lngStraightFlashCt = lngStraightFlashCt + 1
        End If
    Next lngCt

    '数字を比較して手役を決める
    If lngStraightFlashCt = HANDS_MAX Then          'ストレートフラッシ
ュフラグが5の場合
        procRoleCheck = ROLE_STRAIGHT_FLASH '手役をストレートフラッシュ
に設定
        For lngCt = 1 To HANDS_MAX
            Select Case Hands(lngCt).udtHand.CardPoint '手持ちカードが
10,11,12,13,1の場合
            Case Is = 1, 10, 11, 12, 13
'lngStraightFlashCtを1ずつ減らす
                lngStraightFlashCt = lngStraightFlashCt - 1
            Case Else
                lngStraightFlashCt = HANDS_MAX 'フラグの値を5に戻す
            End Select
        Next lngCt
        If lngStraightFlashCt = 0 Then
'lngStraightFlashCtが0ならば
            procRoleCheck = ROLE_LOYAL_STRAIGHT_FLASH '手役をロイヤルス
トレートフラッシュに設定
        End If
        Exit Function
    ElseIf lngFlashCt = HANDS_MAX Then     'フラッシュフラグが5の場合
        procRoleCheck = ROLE_FLASH '手役をフラッシュに設定
        Exit Function
    ElseIf lngStraightCt = HANDS_MAX Then     'ストレートフラグが5の場合
        procRoleCheck = ROLE_STRAIGHT '手役をストレートに設定
        Exit Function
    End If

    Select Case lngPairCt 'ペアフラグのポイント合計値で手役を判別
    Case Is = 2
        procRoleCheck = ROLE_ONE_PAIR    'ワンペアに設定
    Case Is = 3
        procRoleCheck = ROLE_THREE_CARDS 'スリーカードに設定
    Case Is = 4
        procRoleCheck = ROLE_FOUR_CARDS  'フォーカードに設定
    Case Is = 6
        procRoleCheck = ROLE_TWO_PAIR    'ツーペアに設定
    Case Is = 7, 8
        procRoleCheck = ROLE_FULL_HOUSE  'フルハウスに設定
    Case Else
        procRoleCheck = NO_ROLE          '役なしに設定
    End Select
End Function

'COMの手役選択プロシージャ
Private Sub procRoleChoice(Hands() As typHandCard)
    Dim udtChoiceRole As typRole           '交換カード選択方法を決定する
フラグ
    Dim lngCt As Long                 'ループカウント
    Dim lngPairCt As Long             'ペアフラグの個数カウント
    Dim lngFlashCt As Long            'フラッシュフラグの個数カウント
    Dim lngStraightCt As Long         'ストレートフラグの個数カウント
    Dim lngStraightFlashCt As Long 'ストレートとフラッシュにチェックがあ
る個数カウント

    'カードの各フラグの確認
    For lngCt = 1 To HANDS_MAX
        lngPairCt = Hands(lngCt).PairFlg + lngPairCt
        lngFlashCt = Hands(lngCt).FlashFlg + lngFlashCt
        lngStraightCt = Hands(lngCt).StraightFlg + lngStraightCt
```

```
        If (Hands(lngCt).FlashFlg And Hands(lngCt).StraightFlg) = 1 Then
            lngStraightFlashCt = lngStraightFlashCt + 1
        End If
    Next lngCt

    '交換カードの選択方法の決定
    'フラッシュ、ストレート及びストレートフラッシュはフラグが4枚以上ない
場合は狙わない
    If lngStraightFlashCt >= 4 Then
        udtChoiceRole = ROLE_STRAIGHT_FLASH 'ストレートフラッシュ役の場
合
    ElseIf lngFlashCt >= 4 Then
        udtChoiceRole = ROLE_FLASH 'ストレートフラッシュ役の場合
    ElseIf lngStraightCt >= 4 Then
        udtChoiceRole = ROLE_STRAIGHT 'ストレートフラッシュ役の場合
    Else
        Select Case lngPairCt
        Case Is >= 2
            udtChoiceRole = ROLE_ONE_PAIR 'ペア役の場合
        Case Is < lngFlashCt
            udtChoiceRole = ROLE_FLASH 'ストレートフラッシュ役の場合
        Case Is < lngStraightCt
            udtChoiceRole = ROLE_STRAIGHT 'ストレートフラッシュ役の場合
        Case Else
            udtChoiceRole = NO_ROLE '役なしの場合
        End Select
    End If

    For lngCt = 1 To HANDS_MAX
        Select Case udtChoiceRole '枚数の組み合わせから交換するカードを
決定
        Case Is = ROLE_STRAIGHT_FLASH 'ストレートフラッシュ役の場合
            'フラッシュとストレートのフラグのどちらかが0の場合
            If (Hands(lngCt).FlashFlg And Hands(lngCt).StraightFlg) = 0
Then
                Hands(lngCt).ChangeFlg = 1
            End If
        Case Is = ROLE_FLASH 'フラッシュ役の場合
            If Hands(lngCt).FlashFlg = 0 Then 'フラッシュフラグが0の場合
                Hands(lngCt).ChangeFlg = 1
            End If
        Case Is = ROLE_STRAIGHT 'ストレート役の場合
            If Hands(lngCt).StraightFlg = 0 Then 'ストレートフラグが0の
場合
                Hands(lngCt).ChangeFlg = 1
            End If
        Case Is = ROLE_ONE_PAIR 'ペア役の場合
            If Hands(lngCt).PairFlg = 0 Then 'ペアフラグが0の場合
                Hands(lngCt).ChangeFlg = 1
            End If
        Case Is = NO_ROLE '手役なしの場合
            Hands(lngCt).ChangeFlg = 1
        End Select
    Next lngCt
End Sub

'手役の名前をチェックするプロシージャ
Private Function procRoleName(Role As typRole) As String
    Select Case Role 'チェックした手役から名前を判別
    Case Is = NO_ROLE
        procRoleName = "役なし"
    Case Is = ROLE_ONE_PAIR
        procRoleName = "ワンペア"
    Case Is = ROLE_TWO_PAIR
        procRoleName = "ツーペア"
    Case Is = ROLE_THREE_CARDS
        procRoleName = "スリーカード"
    Case Is = ROLE_FOUR_CARDS
        procRoleName = "フォーカード"
    Case Is = ROLE_FULL_HOUSE
        procRoleName = "フルハウス"
    Case Is = ROLE_STRAIGHT
        procRoleName = "ストレート"
    Case Is = ROLE_STRAIGHT_FLASH
        procRoleName = "ストレートフラッシュ"
    Case Is = ROLE_FLASH
        procRoleName = "フラッシュ"
    Case Is = ROLE_LOYAL_STRAIGHT_FLASH
        procRoleName = "ロイヤルストレートフラッシュ"
    End Select
End Function
```

```
'カード表示用プロシージャ
Private Sub procDisplay()
    If mlngDispFlg = 0 Then 'COMカード表示にチェックが入っていない場合、
裏面を表示
        Image1.Picture = LoadPicture(mudtBack.ImageParh)
        Image2.Picture = LoadPicture(mudtBack.ImageParh)
        Image3.Picture = LoadPicture(mudtBack.ImageParh)
        Image4.Picture = LoadPicture(mudtBack.ImageParh)
        Image5.Picture = LoadPicture(mudtBack.ImageParh)
    Else 'COMカード表示にチェックが入っている場合、カード内容を表示
        Image1.Picture = LoadPicture(mudtCOM(1).udtHand.ImageParh)
        Image2.Picture = LoadPicture(mudtCOM(2).udtHand.ImageParh)
        Image3.Picture = LoadPicture(mudtCOM(3).udtHand.ImageParh)
        Image4.Picture = LoadPicture(mudtCOM(4).udtHand.ImageParh)
        Image5.Picture = LoadPicture(mudtCOM(5).udtHand.ImageParh)
    End If
    'ユーザカードの表示
    Image6.Picture = LoadPicture(mudtUser(1).udtHand.ImageParh)
    Image7.Picture = LoadPicture(mudtUser(2).udtHand.ImageParh)
    Image8.Picture = LoadPicture(mudtUser(3).udtHand.ImageParh)
    Image9.Picture = LoadPicture(mudtUser(4).udtHand.ImageParh)
    Image10.Picture = LoadPicture(mudtUser(5).udtHand.ImageParh)
End Sub

'勝ち負け判定用プロシージャ
Private Sub procMatch()
    Dim udtCOMRole As typRole   'COMの手役変数
    Dim udtUserRole As typRole 'プレイヤーの手役変数
    Dim strCOMRole As String    'COMの手役名変数
    Dim strUserRole As String 'プレイヤーの手役名変数
    Dim strMSG As String        'メッセージ保存変数

    '手役の決定
    udtCOMRole = procRoleCheck(mudtCOM)
    udtUserRole = procRoleCheck(mudtUser)

    '各手役を表示させるラベルに手役を表示
    strCOMRole = procRoleName(udtCOMRole)
    strUserRole = procRoleName(udtUserRole)
    Label3.Caption = strCOMRole
    Label4.Caption = strUserRole

    'カード交換ボタンと勝負ボタンを使用不可にする
    CommandButton2.Enabled = False

    mlngDispFlg = 1  'COMカード表示フラグを1にする
    Call procDisplay 'カードを表示

    '手役を比較して表示メッセージを決定
    If udtCOMRole > udtUserRole Then
        strMSG = strCOMRole & "でCOMの勝ち!!!"
    ElseIf udtCOMRole < udtUserRole Then
        strMSG = strUserRole & "でプレイヤーの勝ち!!!"
    ElseIf udtCOMRole = udtUserRole And udtCOMRole <> NO_ROLE Then
        strMSG = "引き分け!!!"
    ElseIf udtCOMRole = udtUserRole And udtCOMRole = NO_ROLE Then
        strMSG = "どちらも役がありません…"
    End If

    MsgBox strMSG 'メッセージボックスを表示

    CommandButton1.SetFocus 'ゲーム開始ボタンにフォーカスをあてる

    Me.Repaint 'UserFormの再描画
End Sub

'手札クリック時の動作プロシージャ
Private Sub procImageClick(Hands As typHandCard, ImageName As Object)
    ImageName.Visible = False   'Imageを非表示にする
    If Hands.ChangeFlg = 0 Then 'カード交換フラグが0の場合
        Hands.ChangeFlg = 1 'カード交換フラグを1にする
        ImageName.Picture = LoadPicture(mudtBack.ImageParh) 'カードの裏側
画像を表示
    Else
        Hands.ChangeFlg = 0 'カード交換フラグを0にする
        ImageName.Picture = LoadPicture(Hands.udtHand.ImageParh) '現在の
手札の画像を表示
    End If
    ImageName.Visible = True 'Imageを表示する
End Sub
```

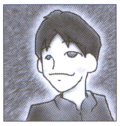
教科「情報」など授業にも使える！

Visual Basic アルゴリズム講座

文・プログラム
／東 幸太

第11回 プロシージャの機能：参照渡しと値渡し

プロシージャとローカル変数を組み合わせることで，プログラムを機能別にまとめることができるようになります。プロシージャの機能を使ってプログラムを分割し，ローカル変数を活用することで変数をとおした依存関係を解消していくわけです。しかし，プロシージャの問題として，返値が一つしかない，という点があります。
今回は，複数の返値をサポートするための機構について説明します。

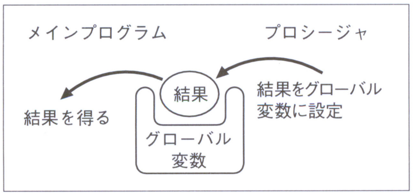


年末は毎年慌ただしいが，一段落ついてしまえば正月までの間は心が落ち着く。この間にプログラムを作成すると，落ち着いているからか洗練されたプログラムができることが多い。重要な部分は年末年始に作るのが筆者の得意技になりつつある。

一つだけの返値

プロシージャは宣言する際に，引数を好きなだけ並べてやることができ，個数の制限は事実上ありません。必要なだけ引数を設定することができます。しかし，返値はfunctionプロシージャの場合，一つしか設定できません。このため，複数の返値をサポートするためには，プロシージャ内部の結果を，グローバル変数を経由してメインプログラムに戻す必要があります（**第1図**）。

しかし，このような用途でグローバル変数を使ってしまうと，プロシージャとメインプログラムが，グローバル変数によってつながってしまいます。プロシージャを変更する際にはメインプログラム側の，メインプログラムを変更する際にはプロシージャ側の，それぞれのグローバル変数の使われ方に注意しなければならなくなってしまいます。これでは，プロシージャをプログラム部品として簡単に利用することはできなくなります。



《第1図》グローバル変数を利用した返値のやり取り

また，返値として配列変数を利用したい場合も，functionプロシージャではグローバル配列変数を中継変数として使うことになり，こちらも同様の問題を抱え込むことになります。

引数の値渡しと参照渡し

複数の返値をサポートするため，VBではプロシージャの引数を「参照渡し」にすることができます。また，「値渡し」にすることもできます。参照渡しと値渡しは，引数として与えられた変数そのものを引数にするのか，その変数の値を引数にするのか，の違いです。どちらのタイプにするかはプロシージャを宣言する際に指定します。

二つの引数をもつプロシージャtestを

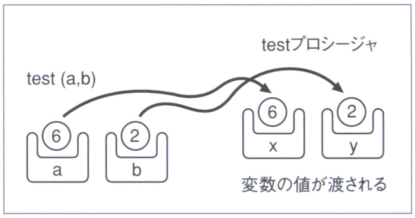
```
Function test!(ByRef x%,ByRef y%)
```

と記述してやると，参照渡しになります。また，

```
Function test!(ByVal x%,ByVal y%)
```

と記述してやると，値渡しになります。指定をしない場合は，参照渡しになります。

数学などで利用する通常の関数は，値渡しになっています。値渡しでは，引数



《第2図》値渡しの引数の動作

の数値がプロシージャに渡されます（**第2図**）。ですから，仮に

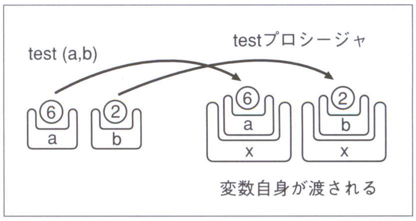
```
c = test(a,b)
```

と記述してプロシージャtestを呼び出した場合，変数aと変数bの値がtestに渡されます。testの内部で変数aと変数bを操作しても，呼び出した側の変数aと変数bには影響は生じません。

一方，参照渡しの場合は，引数そのものがプロシージャに渡されます（**第3図**）。

```
c = test(a,b)
```

と記述してプロシージャtestを呼び出した場合，変数aと変数bがそのままtestに



《第3図》参照渡しの引数の動作

渡されます。このため，test内部で変数aと変数bを操作した場合は，呼び出した側の変数aと変数bが操作されたことになり，影響が生じます。プロシージャの内部の変数aと変数bは，メインプログラム側の変数aと変数bに対する参照となっています。

このように，値渡しであればプロシージャ内部とメインプログラムの間は数値だけがやりとりされるため，相互の関係は薄くなります。それに対し，参照渡しにした場合は，引数として与えたメインプログラムの変数が，プロシージャとの間で共通化されることになり，相互の関係が深くなります。

参照渡しによる複数の返値

参照渡しを利用すると，メインプログラムとプロシージャの間を，メインプログラムの変数によってつなぐことができます（**第4図**）。これを活用すると，プロシージャから複数の返値をメインプログラムに戻すことができます。

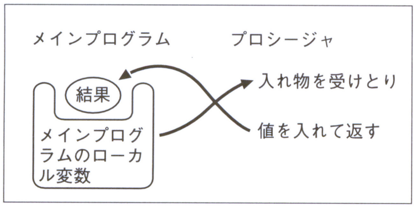
参照渡しのプロシージャtestを以下のように宣言してやります。

```
Function test%(ByRef x1%,ByRef x2%,ByRef x3%,ByRef x4%)
```

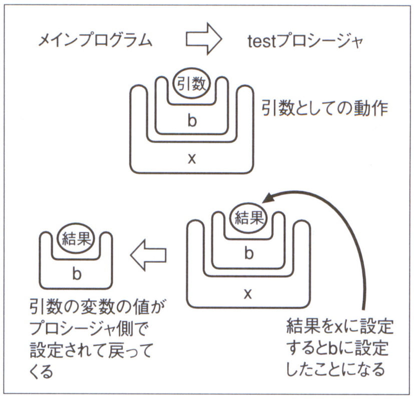
メインプログラムから，

```
a = test (b,c,d,e)
```

として呼び出したとします。test内部では，変数x1から変数x4が引数という形で利用できるわけですが，test内部でこれらの変数に値を代入してやると，メインプログラムの変数bから変数eの値もそれに応じて変化します（**第5図**）。よって，test自体の返値に加えて，引数も返値として利用できるようになります。これによって，testは本来の返値に加えて四つの返値も用意できることになります。

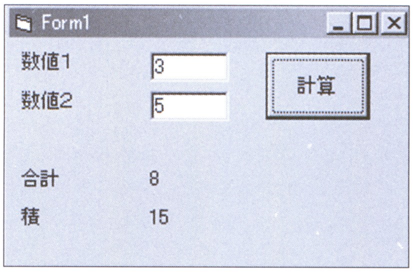


《第4図》参照渡しによる返値のやり取り



《第5図》引数と返値の両方の動作が可能

このように，参照渡しを利用すると，引数がメインプログラムとプロシージャ内部で共通化されるため，返値の個数を自由に増やすことができます。また，引数として配列変数を適用すると，配列変数全体を返値にすることも可能です。

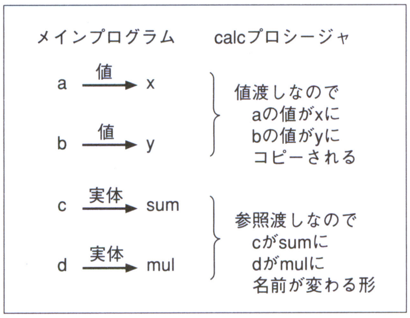


《第6図》リスト1では計算を行うcalcプロシージャの引数を参照渡しにすることで，和と積を同時にメインプログラムに戻している。このように参照渡しを使うことで，複数の返値を実現できる

リスト1は，参照渡しを利用してプロシージャに複数の返値をもたせるサンプルプログラムです。実行すると，**第6図**のような画面が表示されます。二つの数値を入力して，計算ボタンを押すと，二つの数値の合計値と，二つの数値を掛け合わせた値を表示します。**リスト1**には，

《リスト1》参照渡しを利用したサンプルプログラム

```
VERSION 5.00
Begin VB.Form Form1
   Caption         =   "Form1"
   ClientHeight    =   2508
   ClientLeft      =   48
   ClientTop       =   324
   ClientWidth     =   3744
   LinkTopic       =   "Form1"
   ScaleHeight     =   2508
   ScaleWidth      =   3744
   StartUpPosition =   3  'Windows の既定値
   Begin VB.CommandButton Command1
      Caption         =   "計算"
      Height          =   612
      Left            =   2400
      TabIndex        =   8
      Top             =   120
      Width           =   972
   End
   Begin VB.TextBox Text2
      Height          =   264
      Left            =   1320
      TabIndex        =   3
      Text            =   "Text2"
      Top             =   480
      Width           =   732
   End
   Begin VB.TextBox Text1
      Height          =   264
      Left            =   1320
      TabIndex        =   2
      Text            =   "Text1"
      Top             =   120
      Width           =   732
   End
   Begin VB.Label Label6
      Height          =   252
      Left            =   1320
      TabIndex        =   7
      Top             =   1560
      Width           =   972
   End
   Begin VB.Label Label5
      Height          =   252
      Left            =   1320
      TabIndex        =   6
      Top             =   1200
      Width           =   972
   End
   Begin VB.Label Label4
      Caption         =   "積"
      Height          =   252
      Left            =   120
      TabIndex        =   5
      Top             =   1560
      Width           =   972
   End
   Begin VB.Label Label3
      Caption         =   "合計"
      Height          =   252
      Left            =   120
      TabIndex        =   4
      Top             =   1200
      Width           =   972
   End
   Begin VB.Label Label2
      Caption         =   "数値２"
      Height          =   252
      Left            =   120
      TabIndex        =   1
      Top             =   480
      Width           =   972
   End
   Begin VB.Label Label1
      Caption         =   "数値１"
      Height          =   252
      Left            =   120
      TabIndex        =   0
      Top             =   120
      Width           =   972
   End
End
Attribute VB_Name = "Form1"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit

Private Sub Command1_Click()
    Dim a!, b!, c!, d!

    a = Text1.Text
    b = Text2.Text

    calc a, b, c, d 'cとdは参照渡しになるためcal
プロシージャ内で変更可能

    Label5.Caption = c
    Label6.Caption = d

End Sub

Function calc(ByVal x!, ByVal y!, ByRef sum!,
ByRef mul!)
    sum = x + y
    mul = x * y
End Function

Private Sub Form_load() '初期設定
    Text1.Text = ""
    Text2.Text = ""
    Label5.Caption = ""
    Label6.Caption = ""
End Sub
```



《第7図》calcプロシージャの引数

この二つの数値を計算するプロシージャcalcが用意されています。返値として，合計値と，掛け合わせた値の二つが必要になっているため，参照渡しを利用して二つの返値を実現しています。calcには四つの引数がありますが，最初の二つは数値を設定する引数で，これらは値渡しになっています。後の二つが参照渡しになっていて，合計値と掛け合わせた値の二つに対応しています。メインプログラムでは，変数cとdがcalcを呼び出す際に参照渡しされます。calc内では，変数cはsumに，変数dはmulにそれぞれ置き換えられ，計算結果はsumとmulに格納されます。この時，メインプログラム側の変数cと変数dも，それぞれ計算結果が格納され，メインプログラムに戻った後に変数cと変数dを参照することで，結果が得られます（**第7図**）。

subプロシージャ

これまではプロシージャをすべてfunctionプロシージャとして説明してきましたが，VBにはこれ以外にsubプロシージャがあります。subプロシージャは，基本的にはfunctionプロシージャと同じ動作をしますが，直接の返値はありません。よって，通常は返値が不要な場合に利用します。

しかし，参照渡しを利用すれば，subプロシージャでも返値を戻すことが可能となります。よって，機能的には両者の違いはほとんど無くなります。

数学関数的なプロシージャや，返値が一つで済む場合はfunctionプロシージャを使う方が分かりやすいのですが，それ以外の場合はsubプロシージャを使う手もあります。複数の返値がどれも対等な意味をもっているような場合は，subプロシージャ＋参照渡しを利用する方が分かりやすくなります。

subプロシージャの宣言はfunctionプロシージャの宣言と同様，

```
sub test(ByRef a!,ByVal b%)
```

というように記述します。

呼び出す際には，

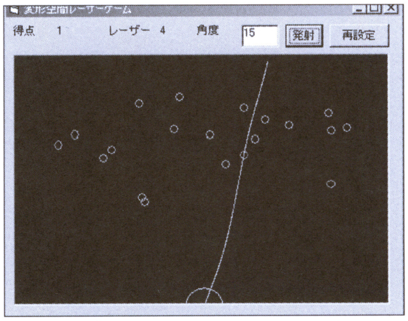
```
test x,10
```

というように，括弧を付けずに引数を並べます。見かけはtestという命令に，xと10というパラメータを指定しているかのようになります。

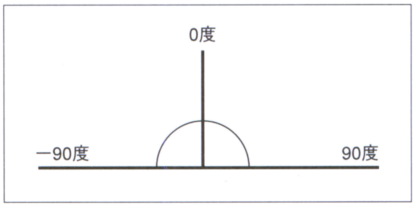
変形空間レーザーゲーム

それでは，参照渡しのプロシージャを利用したゲームをプレイしてみましょう。**リスト2**は変形空間レーザーゲームです。

リスト2を実行すると，**第8図**のような画面が表示されます。ゲームフィールドにはターゲットとなる赤い円が表示されています。また，フィールド下辺中央には砲台があり，プレイヤーはここからレーザーを発射します。レーザーはダイアログに角度を入力し，発射ボタンを押すと発射されます。角度は真上が0度，真右が90度，真左が−90度となっています（**第9図**）。ゲームフィールドには特殊な電磁場が形成されているため，レーザーは真っ直ぐには飛ばず，曲がります。レーザーを5回発射し，最終的に破壊できたターゲットの数が得点となります。



《第8図》変形空間レーザーゲーム。レーザーの軌道は変形空間の影響で変化するため，最初はターゲットが多い方向を適当に狙うしかない。空間の歪み具合が把握できたら，角度を微調整してターゲットを狙っていこう



《第9図》レーザーの角度

リスト2には，convxyという座標変換を行うプロシージャが組み込まれています。画面表示を行う際にこのプロシージャを呼び出すことにより，座標を変化させ，ゲームフィールドの中を見かけ上レーザーが真っ直ぐには飛ばないようにしています。平面の座標はX座標とY座標の二つのパラメータが必要ですから，変化させる場合もX座標とY座標を両方変化させたいところです。この部分をプロシージャにする場合，プロシージャの返値は一つしかありませんから，変換後の座標を二つ返すためには参照渡しによる引数を用意し，この引数に変化後の座標を格納する必要があります。convxyは，二つの引数XとYをもっていますが，この両方が参照渡しになっていて，convxyを呼び出した結果もこの二つに格納するようになっています。

リスト2にはconvxyとは別に，puttargetというターゲットを表示するプロシージャも組み込まれています。こちらは座標に関係する引数を値渡ししています。これを参照渡しにした場合はどうなってしまうか，考えてみてください。

convxyで実行している座標変換は，三角関数を利用して座標の間隔を適宜伸ばしたり縮めたりしています。xaとyaは伸び縮みの度合いを，xpとypは伸びるエリアから縮むエリアの間隔をそれぞれ指定しています。setconditionプロシージャ内部で指定しているこれらの値を変化させると，座標の変形具合が変わります。

終わりに

プログラムをブロック化する際に，プロシージャから複数の返値をとりたい場合があります。これを，グローバル変数を利用せずに，参照渡しによって複数の返値を用意することで，グローバル変数

《リスト2》変形空間レーザーゲームのプログラム・リスト

```
VERSION 5.00
Begin VB.Form Form1
   Caption         =   "変形空間レーザーゲー
ム"
   ClientHeight    =   4836
   ClientLeft      =   48
   ClientTop       =   324
   ClientWidth     =   6432
   LinkTopic       =   "Form1"
   ScaleHeight     =   4836
   ScaleWidth      =   6432
   StartUpPosition =   3  'Windows の既定値
   Begin VB.TextBox Text1
      Height          =   372
      Left            =   3840
      TabIndex        =   1
      Text            =   "Text1"
      Top             =   120
      Width           =   612
   End
   Begin VB.CommandButton Command2
      Caption         =   "発射"
      Height          =   372
      Left            =   4560
      TabIndex        =   2
      Top             =   120
      Width           =   612
   End
   Begin VB.CommandButton Command1
      Caption         =   "再設定"
      Height          =   372
      Left            =   5280
      TabIndex        =   3
      Top             =   120
      Width           =   972
   End
   Begin VB.PictureBox Picture1
      AutoRedraw      =   -1  'True
      Height          =   4092
      Left            =   120
      ScaleHeight     =   4044
      ScaleWidth      =   6084
      TabIndex        =   0
      Top             =   600
      Width           =   6132
   End
   Begin VB.Label Label5
      Caption         =   "角度"
      Height          =   372
      Left            =   3120
      TabIndex        =   8
      Top             =   120
      Width           =   612
   End
   Begin VB.Label Label4
      Height          =   372
      Left            =   2520
      TabIndex        =   7
      Top             =   120
      Width           =   372
   End
   Begin VB.Label Label3
      Caption         =   "レーザー"
      Height          =   372
      Left            =   1680
      TabIndex        =   6
      Top             =   120
      Width           =   852
   End
   Begin VB.Label Label2
      Height          =   372
      Left            =   840
      TabIndex        =   5
      Top             =   120
      Width           =   612
   End
   Begin VB.Label Label1
      Caption         =   "得点"
      Height          =   372
      Left            =   120
      TabIndex        =   4
      Top             =   120
      Width           =   612
   End
End
Attribute VB_Name = "Form1"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit
Dim xsize!, ysize! 'ピクチャボックスのサイズ
Dim score% '得点
Dim leftlaser% 'レーザー残り発射回数
Dim xa!, ya! '座標変換ゲイン
Dim xp!, yp! '座標変換位相ゲイン
Dim xpb!, ypb! '座標変換位相オフセット
Dim tx!(19), ty!(19) 'ターゲット座標
Dim tf%(19) 'ターゲット状態フラグ

Private Sub Command1_Click()
    setcondition
End Sub

Sub setcondition()
    Dim i%
    Picture1.Line (0, 0)-(xsize, ysize), RGB(0,
0, 0), BF 'ピクチャボックスクリア
    xa = xsize / 50 * (Rnd(1) + 0.3) '変形パラメ
ータ決定
    ya = ysize / 50 * (Rnd(1) + 0.3)
    xp = 10 * 2 * 3.14 / xsize * (Rnd(1) + 0.1)
    yp = 10 * 2 * 3.14 / ysize * (Rnd(1) + 0.1)
    xpb = Rnd(1) * 3.1416
    ypb = Rnd(1) * 3.1416
    For i = 0 To 19
        tx(i) = Rnd(1) * xsize * 0.8 + xsize *
0.1 'ターゲット座標設定
        ty(i) = Rnd(1) * ysize * 0.5 + ysize *
0.1
        tf(i) = 1 '存在フラグ
        puttarget tx(i), ty(i), RGB(0, 255, 0) '
ターゲット表示
    Next
    Picture1.Circle (xsize / 2, ysize), xsize /
20, RGB(255, 255, 0) '砲台表示
    leftlaser = 5 'レーザー残数
    score = 0
    Label2.Caption = score
    Label4.Caption = leftlaser
    Picture1.Refresh
End Sub

Private Sub Command2_Click()
    Dim ar!
    Dim lx!, ly!
    Dim nlx!, nly!
    Dim dlx!, dly!
    Dim ntx!, nty!
    Dim c!
    Dim f%, i%

    If leftlaser < 1 Then Exit Sub 'レーザーが無ければ何もしない
    leftlaser = leftlaser - 1 'レーザー残数を減らす
    Label4.Caption = leftlaser

    ar = Val(Text1.Text) / 180 * 3.1416
    lx = xsize / 2 'レーザー初期位置
    ly = ysize
    c = xsize / 1000
    dlx = Sin(ar) * c 'レーザー移動量
    dly = -Cos(ar) * c
    f = 0
    Do While f = 0
        lx = lx + dlx 'レーザー移動
        ly = ly + dly
        nlx = lx 'レーザーの位置の座標変換
        nly = ly
        convxy nlx, nly '座標変換
        For i = 0 To 19
            If tf(i) > 0 Then '存在しているターゲットについて衝突チェック
                ntx = tx(i) 'ターゲットの位置の座標変換
                nty = ty(i)
                convxy ntx, nty '座標変換
                If (ntx - nlx) * (ntx - nlx) + (nty - nly) * (nty - nly) < (xsize / 100) * (xsize / 100) Then
                    tf(i) = 0 '消滅
                    score = score + 1
                    Label2.Caption = score
                    puttarget tx(i), ty(i), RGB(255, 0, 255)
                End If
            End If
        Next
        Picture1.PSet (nlx, nly), RGB(0, 255, 255) 'レーザー表示
        Picture1.Refresh
        If lx < 0 Or ly < 0 Or lx > xsize Or ly > ysize Then f = 1 'レーザーがゲームフィールド外へ出たら終了
    Loop
End Sub

Private Sub Form_load()
    xsize = Picture1.Width 'ピクチャボックスのサイズを取得
    ysize = Picture1.Height
    Label2.Caption = ""
    Label4.Caption = ""
    Text1.Text = ""
    setcondition
End Sub

Sub puttarget(ByVal x!, ByVal y!, c&)
    convxy x, y '座標変換
    Picture1.Circle (x, y), xsize / 100, c
End Sub

Sub convxy(ByRef x!, ByRef y!)
    x = x + xa * Sin(x * xp + xpb) '座標をずらす
    y = y + ya * Sin(y * yp + ypb)
End Sub
```

リスト2の変数表

グローバル変数	
xsize	ピクチャボックス幅
ysize	ピクチャボックス高さ
score	得点
leftlaser	レーザー残り発射回数
xa	座標変換ゲイン（X座標）
ya	座標変換ゲイン（Y座標）
xp	座標変換位相ゲイン（X座標）
yp	座標変換位相ゲイン（Y座標）
xpb	座標変換位相オフセット（X座標）
ypb	座標変換位相オフセット（Y座標）
tx(19)	ターゲットX座標
ty(19)	ターゲットY座標
tf(19)	ターゲット状態フラグ
Command2_Click	プロシージャ内ローカル変数
ar	レーザー進行方向（ラジアン単位）
lx	レーザーX座標
ly	レーザーY座標
nlx	座標変換後のレーザーX座標
nly	座標変換後のレーザーY座標
dlx	X方向レーザー移動量
dly	Y方向レーザー移動量
ntx	座標変換後のターゲットX座標
nty	座標変換後のターゲットY座標

を利用する必要がなくなります。これによって，プロシージャのブロック化がより明確になり，プロシージャを部品的に利用することができるようになります。

頻繁に利用するプロシージャで複数の返値が必要な場合は参照渡しを活用し，プロシージャの部分をそのままコピーすれば，すぐに他のプログラムでも使えるようにしておくと便利です。

続！ベーシックプログラミング教室 BM基礎講座

第4回 「グラフィックにチャレンジ！」

講師紹介

川上朝幸（Kawakami Tomoyuki）
1971年8月31日　大阪府生まれ

93年（株）カプコン入社。サウンドエンジニアとして，「ストリートファイター２」「ヴァンパイア」「ロックマン」シリーズをはじめ「バイオハザード」など，多くのサウンド制作を経て00年退職，翌01年「スタジオ スターノーツ」を起業。サウンドエンジニアとしての活動を軸にスクール講師，プログラミング，ホームページ，ビデオ制作や，カメラ撮影，アマチュアバンドのアルバムプロデュース，自身のバンド「スターノーツ」のライブなど幅広く活動している。

STARNOTES
http://www.starnotes.jp
basic@starnotes.jp

この講座は，ベーシックとして基本的なN88BASICに基づいた内容ですのであらゆるベーシックで応用することができます。

講座内で使用するソフトは，Windows上で実行することができ，容易に入手することができるフリーソフトを利用します。

☆フリーで登録されているベーシックプログラミング・ソフトをいくつか紹介します。

●N88互換BASIC for Windows95
<http://www.vector.co.jp/soft/win95/prog/se055956.html>

●ActiveBasic
<http://www.vector.co.jp/soft/win95/prog/se117283.html>

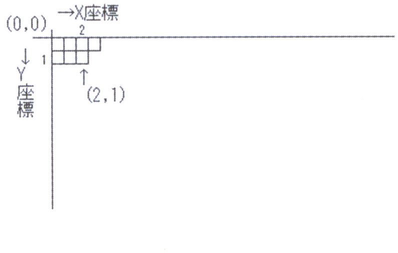
講座内ではこれらを互換BASICと呼ぶことにします。

新年明けましておめでとうございます。2003年になりましたねー，わがスターノーツにとって，今年は勝負の年！全力で突っ走っていきます！！みなさんの今年の目標は何ですか？ぜひオリジナルゲームを作ってみましょう！スケールは大きくなくてもOK，簡単なものでも完成させることで自信に変わっていきます。小さなことからコツコツと！！（笑）今月もがんばっていきましょう！

グラフィック画面をおぼえる

ベーシックでサウンドノベルを作ろう！ということで，折り返し地点の4回目になりました。今回はグラフィックに関連する命令と，互換ベーシックに搭載されているBMP（ビットマップ）画像を表示する機能を使って，よりゲームを盛り上げていきましょう！

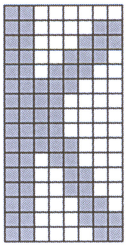
グラフィックに関連する命令語を説明する前に，BASICで扱う画面について簡単に説明します。これまではPRINT命令とLOCATE命令を組み合わせて文字を表示していました。この時の画面をテキスト画面といいます。左上を原点（0,0）として横がX座標，縦がY座標になっています。



座標は半角文字単位になっています。数学との違いはY座標です。下に向かって座標が大きくなります。このときマイナス記号はつきません。

画面に線を引いたり，色を塗ったりする画面をグラフィック画面といいます。この画面もテキスト画面と同じく，左上を原点（0,0）として横がX座標，縦がY座標になっています。

グラフィック画面では，テキスト画面とは違いドット単位で座標を表します。ドットとは画面に表示できる最小の点のことをいいます（ピクセルともいいます）。



半角文字は，横8×縦16ドットで表されています。全角文字は横16×縦16ドットになります。

グラフィックとテキストをうまく表示するには，これを知っておくと大変便利です。

互換BASICでは，実行画面の表示領域を広げればそれだけ広く画面を使用することができます。横や縦の座標の範囲は画面のサイズとなります。しかし，あらゆる環境で問題なく動作することを考えると640×400ドット以内で

制作するほうがよいでしょう。

テキスト画面とグラフィック画面について説明しましたが，N88互換BASICでは文字もグラフィックもまとめて一つの画面に表示されます。座標を扱う上でわかりやすいように二つの画面として説明していますので，実際にはそれぞれの画面を独立して操作することはできませんので注意してください。

[ワンポイント]
テキスト画面とグラフィック画面

画面については，使用する環境によって仕様が変わります。たとえば，PC9801などでN88BASICを使用する場合，テキスト画面とグラフィック画面を独立して使用することができます。表示する際はそれぞれの画面を合成したものが表示されますが，実際には別々のウインドウで管理されているため，文字だけをクリアしたり，グラフィックだけを消去するといったことができます。N88互換BASICでは，グラフィック画面にテキストを表示していますので一画面しかもたないことになります。

また画面を使用できる範囲も環境によって異なります。PC9801の場合テキスト画面で80文字25行，グラフィック画面で最大640×400ドットを表示することができます。

実際の使用環境に応じてプログラミングも変わってきますので注意してください。本講座では，互換BASICでの動作確認したプログラムを掲載しています。他の環境でも動作すると思いますが，エラーなどが出るようであれば，お使いの環境について確認してみてください。

※ACTIVE BASICでは，それぞれ独立して使うことができます。N88互換BASICとの互換を取るために，CLS命令を使うときには CLS 3 と指定して実行してください。

グラフィック命令のいろいろ

それではグラフィック命令について説明していきます。

●PSET（X,Y）,C

画面に点を表示する命令です。X,Yは，グラフィック画面のドット座標，Cは表示するカラーの指定になります。色番号を省略した場合は白色になります。

```
0 = 黒
1 = 青
2 = 赤
3 = 紫
4 = 緑
5 = 水色
6 = 黄
7 = 白
```

```
10 PSET(300,200),6
```

と入力して実行してみましょう。画面の中央あたりに小さな黄色い点が表示されましたね。

カラーの番号はCOLOR命令をはじめ，グラフィック命令の色指定の際に使われますので覚えておきましょう。

COLOR命令は，PRINT文などで表示する文字の色を変えることができます。一度指定すると次に別の色を指定するまで保持されます。

```
10 CLS 3:COLOR 5
20 PRINT "TEST "
30 COLOR 7
40 PRINT "TEST "
```

PSETで点を描く方法を工夫することで点を動かすことができます。流れ星のような効果ですね。次のプログラムを入力して実行してみてください。

```
10 CLS 3 '流れ星
20 FOR I=1 TO 300
30    PSET(I,I),0
40    PSET(I+1,I+1),6
50 NEXT I
```

星が小さいので見にくいですが，左上から右下に向けて一瞬のうちに星が流れていますね。30行目で黒い点を描くことで，すでに表示されている点を消す仕組みになっています。消した後に40行目で新たな点を描いています。この繰り返しで移動しているように見えるわけですね。

●LINE(X1,Y1)-(X2,Y2),C,BまたはBF,C2

画面に直線や四角を描く命令です。始点(X1,Y1)と終点(X2,Y2)を結ぶ直線を描きます。Cはカラーの指定です。Bを追加指定すると2点を対角線とする四角形を描きます。BFを追加指定するとC2で指定したカラーで内部を塗りつぶします。Cが枠になります。C2を省略するとCと同じ色で塗りつぶします。

次のプログラムを実行してLINE命令の使い方を確認してください。

```
10 CLS 3
20 LINE(10,10)-(100,20)
30 LINE(10,50)-(100,60),6
40 LINE(10,100)-(100,150),5,B
50 LINE(10,170)-(100,250),5,BF,3
```

●CIRCLE（X,Y）,R,C,,,,F,C2

円を描く命令です。ここでは基本的な使い方について解説します。

X,Yは円の中心座標，Rは円の半径をドット数で指定します。Cは表示するカラーの指定になります。Fを追加指定するとC2で指定したカラーで内部を塗りつぶします。Cが枠になります。C2を省略するとCと同じ色で塗りつぶします。

次のプログラムを実行してCIRCLE命令の使い方を確認してください。

```
10 CLS 3
20 CIRCLE(60,60),50
30 CIRCLE(120,120),50,5
40 CIRCLE(180,180),50,5,,,,F,3
```

CIRCLE命令には，楕円や扇形を描く機能もあります。詳しくは，互換BASICのHELPなどを参考にしてください。

●PAINT（X,Y）,C,C2

PAINT命令は，線や円で囲まれた部分を塗りつぶす命令です。塗りつぶしたい部分の１点をX,Y座標で指定し，Cで塗りつぶす色を指定，C2で境界とな

る色を指定します。ただし線や円で囲まれた部分の境界線はすべて同じ色である必要があります。

次のプログラムを実行してPAINT命令の使い方を確認してください。

```
10 CLS 3
20 LINE (20,50)-(120,50),5
30 LINE -(80,150),5
40 LINE -(20,50),5
50 PAINT (50,70),2,5
```

50行目のPAINT命令の座標 (50,70) は，20～40行目で描いた三角形の内部の点です。内部であれば座標の指定はどこでもかまいません。ここで20～40行目のLINE命令の使い方に注目してみましょう。20行目は始点と終点の両方を指定していますが，30,40行目では，-（ハイフン）から始まって終点しか指定していません。これは前回の終点の座標が記憶されているからです。この方法を使うとLINE命令の使い方が随分楽になりますね。

続いてBMP（ビットマップ）形式の画像を表示する方法です。互換BASICでは，WINDOWSで実行されている利点を利用して，WINDOWSの標準画像形式のBMPファイルを扱うことができます。

BLOAD "ファイル名",変数名
PUT@ (X,Y), 変数名,PSET,,,<横倍率>,<縦倍率>

BLOAD命令で画像をメモリに読みこみ，PUT@命令で，指定の座標位置に画像を表示します。倍率を指定することで，表示サイズを変えることもできます。

次のプログラムを実行してBLOAD命令とPUT@ 命令の使い方を確認してください。

```
10 CLS 3
20 LOCATE 10,5:PRINT "ビットマップ画像表示"
30 BLOAD "starnotes.bmp",A
40 PUT@ (100,100),A,PSET
```

ファイル名がうまく指定されていないと表示されません。互換BASICの実行ディレクトリを確認して指定してください。パスの指定もできます。
"d:¥bm¥starnotes.bmp"

次に同じ画像を縮小して重ねて表示してみましょう。次の一行を追加して実行してください。

```
50 PUT@ (150,125),A,PSET,,,0.5,0.5
```

拡大縮小の比率によっては画像きれいに表示されない場合があります。いろいろ試してみてください。PUT@ 命令は，PSETと指定している部分を変えることで画像の表示の仕方を変えることができます。2002年12月号の星音高校のコーナーでビットマップのマスク処理について説明しています。その他，詳しくは互換BASICなどのHELPを参照してください。

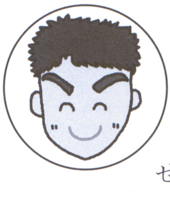
基本的なグラフィック命令について説明してきましたがいかがでしたか？ゲーム画面を飾る上で重要な命令ですが，けっこう手間もかかり慣れるのに時間がかかると思います。これらの命令を使うにあたってのポイントは座標の管理にあります。画面を作るときに，方眼紙などを用意して画面に見立てて下書きをします。その後，方眼紙の座標をチェックしプログラムに入力していけば作業もスムーズにできると思います。ビットマップの作成の際にも役に立ちますよ。

サンプルプログラムを入力しよう！

（ここで使用しているビットマップファイルはホームページよりダウンロードできます。）

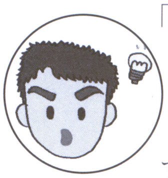
```
100 '星音高校の謎　サウンドノベル版
110 '                2003.1.8   Ver0.03
120 '/********** 初期設定 **********
130 DIM MESSAGE$                    '画面に表示するメッセージの一時変数
140 GYO=14                          '画面に表示する行数
150 X=3:Y=6:YM=23                   'シナリオ表示座標/最下行
160 XG=50:YG=64:XGT=100:YGT=10      'グラフィック座標/タイトル座標
170 H=0                             '画面切り替えメッセージ非表示用フラグ
180 G=99                            '表示する背景番号（99のときは非表示）
190 DIM GFILE$(3)                   'グラフィックファイル名保存用配列
200 '/******* データ読み込み *******
210 RESTORE *GFILE
220 FOR I=0 TO 3:READ GFILE$(I):NEXT I
230 '/******* メインルーチン *******
240 RESTORE *000:G=99:GOSUB *STORY                 '000 OPENING
250 RESTORE *001:GOSUB *STORY                      '001 OPENING B
260 RESTORE *002:GOSUB *STORY                      '002 SEIMON
270  GOSUB *KEYIN                                  '002 POINT01-SEIMON
280  IF A$="1" THEN RESTORE *00201 ELSE RESTORE *00202
290  GOSUB *STORY
300 RESTORE *003:H=2:G=1:GOSUB *STORY              '003 GROUND
310 RESTORE *004:GOSUB *STORY                      '004 GROUND
320  GOSUB *KEYIN                                  '004 POINT02-GROUND
330  IF A$="1" THEN RESTORE *00401 ELSE RESTORE *00402
340  GOSUB *STORY
350 RESTORE *005:H=2:G=2:GOSUB *STORY              '005 KYOSHITU
360 RESTORE *006:H=2:G=3:GOSUB *STORY              '006 廊下1
370 END
380 '/********** 画面表示 **********
390 *STORY
400    CLS 3
410    BLOAD GFILE$(0),G1:PUT@ (XGT,YGT),G1,PSET      'TILTE表示
420    IF G<99 THEN BLOAD GFILE$(G),G1:PUT@ (XG,YG),G1,PSET
430    IF H=2 THEN GOSUB *HITS:H=0
```

```
440    FOR I=0 TO GYO
450       READ MESSAGE$:Y1=I
460       IF MESSAGE$="E" THEN MESSAGE$= "":I=GYO
470       IF MESSAGE$="EH" THEN MESSAGE$= "":I=GYO:H=1
480       LOCATE X,Y+Y1:PRINT MESSAGE$
490    NEXT
500    IF H=1 THEN H=0:RETURN ELSE GOSUB *HITS
510 RETURN
520 *HITS  '/********** HIT[SPACE]KEY **********
530    LOCATE X,YM:PRINT SPACE$(X)+"HIT [SPACE] KEY"
540    *KEYCHECK:IF INKEY$<>" " THEN *KEYCHECK
550 RETURN
560 *KEYIN
570    A$=INKEY$
580    IF A$="1" OR A$="2" THEN RETURN
590 GOTO *KEYIN
600 *GFILE    '/***** グラフィックファイル名 *****
610 DATA "50.BMP","15.BMP","1.BMP","2.BMP"
620 *000       '/****** OPENING ******
630 DATA "2002年，春。入学早々この星音高校にまつわる奇妙なウワサを聞いた，"
640 DATA "クラスメイトの楽人「ガクト」と和音「カズネ」は，そのウワサの真相を"
650 DATA "探るべく動き出すことにした。","E"
660 *001       '/***** OPENING B *****
670 DATA "カズネ「ねえねえ，ガクト，本当に行くよね？」"
680 DATA "ガクト「あ，当たり前だろ？大体この世の中にそんなおかしな話があってたまる"
690 DATA "          もんか。絶対ウワサに決まってる！」"
700 DATA "カズネ「とか何とか言っちゃって～，ちょっと怖いんじゃないのぉ？」"
710 DATA "ガクト「な，な，何言ってんだよ，よ，よし分かった。じゃあ，こ，今晩乗り込"
720 DATA "          もうぜ！」"
730 DATA "カズネ「(笑) じゃあ今晩，正門前に午前0時ね！」","E"
740 *002       '/****** SEIMON *******
750 DATA "カズネ「遅いーっ！もう20分過ぎてるよ！」"
760 DATA "ガクト「すまん，色々と心の準・・調査のための準備に手間取って・・・」"
770 DATA "カズネ「まあ，いいわ。さてと，どうやって入ろうかなあ」",""
780 DATA "  1・・・カズネ「面倒だから，この門からよじ登って入らない？」",""
790 DATA "  2・・・ガクト「となりにある通用口をこじ開けたら中に入れそうだぜ。」",""
800 DATA "HIT [1] or [2] KEY","EH"
810 *00201    '/**** SEIMON 1 *****
820 DATA "ガクト「おまえって，相変わらず大胆だな。」","カズネ「先に行くわよ。よいしょっ」
"
830 DATA "ガクト「お，おい待ってくれよぉ～」","E"
840 *00202    '/**** SEIMON 2 *****
850 DATA "カズネ「大丈夫？本当に開くの？」","ガクト「まかせとけって」", ""
860 DATA "（ガチャッ）", ""
870 DATA "ガクト「ほらっ。」","カズネ「相変わらず器用な人ね・・・」","E"
880 *003        '/***** GROUND *****
890 DATA "こうして二人はグランドに忍び込むことに成功した。","E"
900 *004        '/***** GROUND 2 ****
910 DATA "二人は月の光を背に早足で校舎に向かっていった。",""
920 DATA "ガクト「グランドに入ったはいいが，校内にはどうやって入ればいいんだろう？」",""
930 DATA "  1・・・ガクト「こんなとき意外と非常口が開いてたりするんだよな・・・」",""
940 DATA "  2・・・ガクト「カズネ，なんかいい方法ない？」",""
950 DATA "HIT [1] or [2] KEY","EH"
960 *00401      '/***** GROUND 3 ****
970 DATA "ガクト「う～ん・・・，鍵がかかっていて入れそうにないな・・・」"
980 DATA "カズネ「こっちこっち。」","ガクト「ん？」
990 *00402      '/***** GROUND 4 ****
1000 DATA "カズネ「ふふふ。今日の最後の戸締り，私がやったんだ～」"
1010 DATA "ガクト「あ！ということは・・・」","カズネ「そういうこと。ほら，ここ！」"
1020 DATA "ガクト「あらかじめこの窓だけ開けておいたワケだ・・」",""
1030 DATA "二人は教室に入ることができた。","E"
1040 *005       '/***** ROOM 1 ****
1050 DATA "カズネ「学校で一番あやしい所と言えば？」"
1060 DATA "ガクト「やっぱりトイレか・・・あとは理科室なんかも気味悪そうだし・・」"
1070 DATA "カズネ「じゃあ，まずはトイレにいってみようよ！」","E"
1080 *006        '/***** ROOM 2 ****
1090 DATA "二人は教室を出て，ゆっくりとトイレの方に近づいてみた。","","E"
```

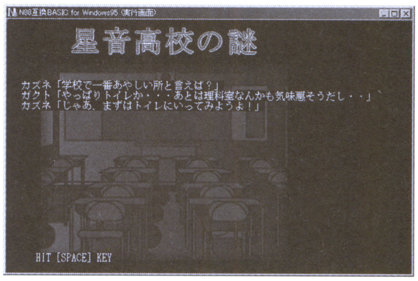


「星音高校パソコン研究部のゆうです。先月号のプログラムにグラフィックを組み合わせてみました。メインルーチンも先月よりすっきりできました！」

「ほんとだね。うまくサブルーチンを利用しているね。ビットマップファイルもうまく表示できてるし，ずいぶんゲームらしくなってきたね。せっかくだからLINE命令とかも使って，画面をもう一工夫してみたらどうかな。来月までの宿題だね。」



「あ！ビットマップに必死で忘れてた・・・宿題ですね，わかりました！がんばります！」



練習問題にチャレンジ

まずは1月号の問題の解答から。

1月号の問題

＜問題＞

X=10，Y=10の座標にアルファベットのA（半角）を表示し，キーボードの4を押すと左に，6を押すと右に半角ずつ移動するプログラムを作りなさい。Aの移動できる座標はX>=2～X<=20とする。　(12月号問題)

12月号の問題で作ったプログラムを次のように改造してください。

1．画面の最上部に，文字Aの現在の座標を表示する。(X=○　Y=○)

2．文字の移動時に座標画面がちらつかないようにする。(CLSを使わない)

3．上下の移動もできるようにする。(Y>=３～Y<=15)

4．サブルーチンを使い，プログラムを効率化する。

解答例

```
10 CLS 3
20 X=10:Y=10:X1=X:Y1=Y
30 GOSUB *HYOJI
40 A$=INKEY$
50     IF A$="4" AND X>2 THEN X=X-1:GOSUB *HYOJI
60     IF A$="6" AND X<20 THEN X=X+1:GOSUB *HYOJI
70     IF A$="2" AND Y<15 THEN Y=Y+1:GOSUB *HYOJI
80     IF A$="8" AND Y>3 THEN Y=Y-1:GOSUB *HYOJI
90 GOTO 40
100 *HYOJI
110     LOCATE X1,Y1:PRINT " "
120     LOCATE X,Y:PRINT "A"
130     LOCATE 0,0:PRINT "X=";X;" Y=";Y;"  "
140     X1=X:Y1=Y
150 RETURN
```

さて今月号の問題です。

＜問題＞

1月号の問題で作ったプログラムでは，"A"という文字を動かしていましたが，これを半径50 で黄色に塗りつぶした円に変更しましょう。円の動ける範囲は，

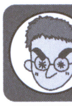
30<X<400，30<Y<300，とします。また移動できる範囲の境界線を白で引いてください。

◆注意

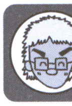
円が半径50であることに注意し，円が移動できる範囲からはみ出さないようにしましょう。

舞台は，星音(ほしね)高校 パソコン研究部。4人の部員たちがベーマガを片手にゲーム制作にチャレンジ！今月も，ゆうちゃんの音についてのお話です。今回はコンピュータミュージックについてのお話です。

登場人物


「ちぃさん」(ちぃ)・・・2年生。部長。VBAにはまりつつある今日この頃。ちょっと戸惑いはあるけど，ちょっとずつ慣れてきました。

「みぃちゃん」(みぃ)・・・2年生。よっさん，ゆうちゃんに刺激され最近歌の練習をはじめたとか。ステージデビューは間近！？


「よっさん」(よっ)・・・3年生。頼りになる先輩。今ホームページ作り夢中!! とてもかっこいいページができあがったようです。

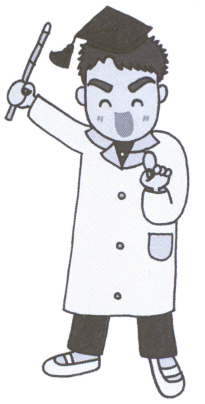
「ゆうちゃん」(ゆう)・・・1年生。最近メキメキと上達してきている期待の1年生！

～星音高校，パソコン研究部部室～

ゆう：「みんなまだかな～。もうすぐ時間なのに・・・おそいなぁ・・・」

ガラガラ・・・（戸が開く）

みぃ：「ゆうちゃん！こんにちは！あれ，どうしたのゆうちゃん。今日はもう準備ばっちりね。」



ゆう：「こんにちは!! 2回目だからちょっとワクワクしてしまって・・・」

みぃ：「で，居ても立っていられなくて早めに準備してた，ってわけね。」

ゆう：「そ，そうなんです。」

みぃ：「今日の講座楽しみね。がんばってねー。」

ガラガラ・・・

ちぃ：「こんにちはー！」

よっ：「お，みんな揃ってるな。ほな早速はじめてもらおうか～。白衣を用意してくるから，ちょっとまっとき・・・って・・・ゆうちゃん，もう白衣着てるやん・・・」

ゆう：「あ，は，はい・・・」

みぃ：「意気込みばっちりよ！」

ちぃ：「ほんとだね。ゆうちゃんがんばれー」

よっ：「・・・」

ゆう：「どうしたんですか？先輩。」

よっ：「いや，今回も出番は最初だけかなと・・・」

みぃ：「何言ってるんですか～！ゆうちゃんを応援してあげてくださいね～。」

ちぃ：「じゃ，はじめようか。ゆうちゃん，よろしく。」

一同：パチパチパチ～（拍手）

ゆう：「今回はコンピュータミュージックについてお話したいと思います。」

よっ：「おれはもうはじめてるで。バンドのデモ作ったりするのに便利なんや。」

ちぃ：「え！そうなんですか～。よっさん先輩，作曲もできるんですか？」

よっ：「ちょっとだけな。どっちかというと歌詞を書く方が得意やけどな。」

みぃ：「よっさん先輩の歌詞ってどんな歌詞なんだろう・・・」

よっ：「愛情たっぷりで情熱的な詞や」

ゆう：「え，えっと・・はじめますね・・・。コンピュータミュージックっていうのは，簡単に言ってしまえばコンピュータで音楽をする！ただそれだけのことです。とはいっても，コンピュータのキーボードをピアノのように華麗な指使いで演奏するわけではありません。人間がコンピュータに演奏の指示を与えることで，コンピュータがその指示どおりに，さまざまな楽器を演奏をしてくれます。つまり，コンピュータで音楽をする！というのは，コンピュータに演奏の指示を与える，コンピュータが演奏するためのデータを制作するということなります。」

ちぃ：「よく"打ち込み"っていう言葉も聞くけど，これはデータを制作ための入力をするっていうことなのかな？」

ゆう：「そうですね。コンピュータミュージックで作ったデータと一緒に演奏するときには，"ドラムは打ち込みでやろう！"みたいな感じで使うときもありますね。」

ちぃ：「なるほど。」

ゆう：「コンピュータミュージックは，卓上（デスクトップ）で音楽を制作するというところから，デスクトップミュージック（Desk Top Music）略してDTMとも呼ばれています。DTMに必要なシステムは，コンピュータとDTM専用ソフトウェア，そして音源モジュールになります。」

みぃ：「卓上音楽ってなんだか面白そうね。」

ゆう：「DTMで使用するソフトウェアは，シーケンサーと呼ばれるもので演奏データの制作をしたり，データをもとに演奏したりします。音源モジュールとは，コンピュータが演奏する楽器がたくさん詰まっている電子楽器です。コンピュータと接続することで，ピアノやバイオリン，サックスなどの音を瞬時に呼びだし演奏することができます。最近の音源モジュールには，1000種類以上の音色が内臓されているので，一台でオーケストラサウンドなども再現できます。また，コンピュータの内部で音源モジュールを再現するソフトウェアシンセサイザーと呼ばれるソフトもあります。」

ちぃ：「けっこう揃えるものがたくさんあるんだね。」

みぃ：「お金かかりそう。」

よっ：「今は初心者向けの入門パックみたいなのもいろいろでるから，5万円以下でひととおり揃うで。」

ゆう：「ここでもうひとつ用語がでてきます。どこかで耳にしたことはあるかと思いますが，DTMでは演奏データを制作したり，パソコンと音源モジュールなどの電子楽器を接続するときに，"MIDI"と呼ばれる規格を使用しています。DTMを理解するためには，このMIDIという規格について知っておく必要があります。」

よっ：「(M) みんな（I）いっしょに（D）どこかに（I）行こう！」

みぃ：「ほっておきましょう・・・」

ゆう：「"MIDI"は「Musical Instrument Digital Interface」の略で「ミディ」と読みます。簡単に日本語にすると「楽器間をデジタル接続する」となります。楽器を演奏したときの情報や楽器を演奏するためのさまざまな情報をデジタル化しデータにすることで，電子楽器同士でコミュニケーションをとることができるようになります。」

みぃ：「うーん。わかったようなわからないような・・・」

ゆう：「そうですね。具体的な例を挙げてみますね。MIDI規格を採用したデジタルピアノとパソコンとの接続を例に説明します。

デジタルピアノ　←MIDIで接続→　パソコン（シーケンサー）

デジタルピアノとパソコンをMIDIで接続し，シーケンサーと呼ばれるアプリケーションを使用することで，デジタルピアノで演奏した情報を記録することができます。「どの鍵盤を」「どの位の強さで」「どの位の長さ」弾いたのか，といった情報をデータにします。このデータをMIDIデータといいます。シーケンサーを使うと記録したMIDIデータをファイルとして保存できるので，いつでもデータを呼び出し演奏を再現することができます。また記録した演奏情報を編集できるので，間違えて演奏した音を簡単に修正することができます。また演奏情報を楽譜にしてプリンタで印刷することもできます。シーケンサーは音楽専用のワープロみたいなものですね。」

みぃ：「よくわかったわ。先月のお話にあった音をデジタル録音するというとは，また違った方法なのね。」

ゆう：「そうなんです。MIDIのデータ量は，オーディオ信号を録音しファイルにした場合（Windowsの.Wavファイルなど）にくらべると圧倒的にデータのサイズが小さくなります。5分の曲でも，MIDIデータにすれば数十KB程度ファイルサイズなので，一瞬のうちにメールで送ることもできます。」

ちぃ：「演奏データがファイルになっているということは，音そのものは含まれていないんだよね。」

ゆう：「はい。そうです。楽器の音は，MIDIデータを再生する環境にセットアップされている音源の音が鳴ります。」

ちぃ：「ということは，環境によって音が変わる可能性もあるんだよね。」

ゆう：「そうですね。ある程度は，音源モジュールの規格（GM規格など）でフォローされていますが，制作時の音源以外で再生した場合は少し音が変わってしまうのは仕方ないですね。とはいえ，うまく使えばとても便利です。街で聞こえる音楽の多くでMIDIが使われています。

[通信カラオケ] 毎月数百曲にもわたる新曲がどんどん入ってくる通信カラオケ。この仕組みは，カラオケボックスにある専用のカラオケ機にMIDI音源が搭載されており，MIDIデータとして配信された曲を再生します。MIDIデータができあがれば全国に瞬時に一斉配信できるので新曲がすぐに歌えるんですね。

[BGM] スーパーや飲食店などに入ったとき，ヒット曲が歌でなく楽器で演奏されているのを聴いたことありませんか？ほとんどの場合がMIDIで作られた曲になっています。よく聴くと実際のCDの曲とは似てるけど違うというのがわかると思います。

[TVゲーム,着メロ] 容量に制限の厳しい媒体なので，ゲームごとや機種ごとに音源を用意し，演奏データを差し替えることで曲を瞬時に切り替えたり，曲を切れ目なくループ（くりかえし）再生したり，多くの曲を再生することがきます。

まだまだ，身近にはたくさんのMIDIを使った音楽が流れています。もちろんプロのレコーディングでもガンガン使われています！」

みぃ：「そうだったんだー。これから音楽の聴き方が変わりそう。」

ちぃ：「次回は，サウンドノベルにMIDIを組み込んでみようか！」

ゆう：「そうしたいのですが，僕が調べた限りでは，ACTIVE BASICでは，WAVファイルやMIDEファイルを演奏することができるのですが，N88互換BASICでは，できそうにないのです・・・」

ちぃ：「え？そうなんだ・・・。よし僕もちょっと調べてみるよ。」

みぃ：「いい感じでゲームできてきてるから，がんばってね！」

ゆう：「はい！がんばります！ということで，コンピュータミュージックのお話をおわります！」

一同：パチパチパチ～（拍手）

この講座で使用しているプログラムデータは下記のホームページでダウンロードすることができます。
http://www.beeplus.or.jp/bm/basic/

「ベーシックプログラミング通信教育講座」
http://www.beeplus.or.jp/bm/basic/kouza.html
では，連載「続！ベーシックプログラミング教室」に関する質問をメールにて受けています。回答は本記事ライターの川上先生です。みなさんの参加をお待ちしています。

☆BM基礎講座「ベーシックプログラミング教室」
制作 スタジオスターノーツ
http://www.starnotes.jp/
企画 有限会社 ビープラス
http://www.beeplus.or.jp/

デジタルの世界へようこそ！

―― コンピュータの不思議をわかりやすく解説 ――

第1回　ANDとORから始めよう

鹿島　洋一（千葉県立八街高等学校教諭）

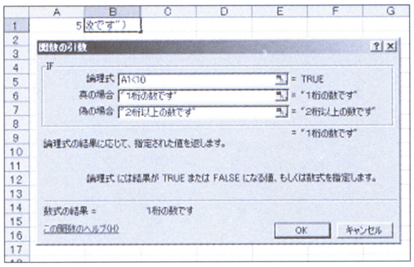
日ごろコンピュータを扱っているなかで，ソフトウェアの操作方法はわかっても，なぜか釈然としなかったことはありませんか。また，プログラム･リストを見ても，なぜそんなふうに書いてあるのかわからなかったことはありませんか。そんなとき，「コンピュータはブラックボックスだから」とあきらめていませんか。

この連載では，コンピュータで扱うデジタルの世界に皆さんをご招待します。2進数？16進数？論理和？など「初級シスアド」や「基本情報処理」などの試験対策としても役に立つ内容です。いろいろなテーマを取りあげ，解説することをとおして，皆さんが今まで感じていた不思議に，納得を与えられたら良いなと思っています。プログラムを書くときのヒントになる話もあるはずです。難しい話は，身近にあるソフトウェアを使って，確認し納得してから先に進めるように書いていくつもりです。一緒に操作しながら読んでみてください。

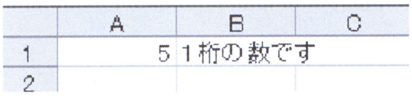
第１回目はANDとORです。まずエクセルを起動してください。さあ，はじめましょう。

IF文は知っているかな

ANDとORの話をするまえに，IF文の話をしておきます。エクセルやBASICでIF文を実行させると，コンピュータは条件が成り立つかどうかを判断して，指定された手順を実行します。



《第1図》セルB1に合わせ，挿入→関数→if関数を選ぶ



《第2図》A1に好きな数値を入れるとB1の表記が変わる

エクセルを起動してみましょう。セルA1には好きな数字を入れてください。セルB1には計算式を入れて，セルA1の数が１桁かどうか判定させてみましょう。

セルB1をクリックしてアクティブにしてから，メニューから「挿入」「関数」を選びます。一覧からif関数を選ぶとダイアログボックスが出てくるので，**第1図**を参考にして入力してください。

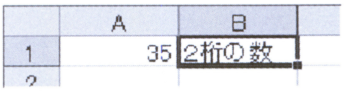
OKをクリックすると，セルB1には =IF(A1＜10,"１桁の数です","２桁以上の数です")　という数式が入力されますので，セルA1に５や10といった数値を入力してみて下さい。セルB1の表示が変わるはずです(**第2図**)。

エクセルでは，

＝IF(条件,処理１,処理２)

の形でセルに数式を記述すると，(条件)が成り立つとき(処理１)を行い，そうでないとき(処理２)を行います。

ANDは知っているかな



《第3図》2桁の判定をさせる

セルA1が２桁の数かどうか判定させてみましょう。セルB1に

=IF(AND(A1>＝10,A1<100),"２桁の数","２桁でない")

と入力します。セルA1の数をいろいろ変えてみてください。セルB1の表示が変わります(**第3図**)。

＝IF((AND(条件１),(条件２)),(処理１),(処理２)) は，(条件１)，(条件２)がともに成り立てば(処理１)を行い，そうでないとき(処理２)を行います。

ORも使ってみよう

ORを使って，セルA1が２桁の数かどうか判定させてみましょう。セルB1の数式を　=IF(OR(A1＜10,A1>＝100),"２桁ではありません","２桁の数です")　と直してみましょう。セルA1の数をいろいろ変えて，正しく判定できるか確かめて下さい。

＝IF((OR(条件１),(条件２)),(処理１),(処理２)) は，(条件１)，(条件２)のどちらかが成り立てば(処理１)を行い，そうでないとき(処理２)を行います。

真(しん)と偽(ぎ)

さて，どこのセルでも構いませんので次の数式を入力してみましょう。

＝IF((3＞5),"真(TRUE)","偽(FALSE)")

このセルは「偽(FALSE)」と表示されます。このときコンピュータはどのようにして，条件が成り立つかどうかを判断して処理を実行しているのでしょうか。新しいワークシートを用意して，

セルA1に　＝(5>3)

セルＡ２に　＝(5<3)と入力します。

それぞれTRUE　FALSEと表示されます。これは「真(成り立つ)」，「偽(成り立たない)」という意味です。

セルB1に　＝A1＊100

セルＢ２に　＝A2＊100

を入力してみてください。それぞれ，100　0　と表示されるはずです(**第4図**)。

	A	B
1	TRUE	100
2	FALSE	0
3		

《第4図》100　0と表示されたかな？

エクセルでは，

TRUE＊100＝100なので，TRUE＝1

FALSE＊100＝0　なので，FALSE＝0

であるとわかります。

条件が成り立つとき，

TRUE　または　1

条件が成り立たないとき，

FALSE　または　0

真理値表

それでは，AND(条件1,条件2）とOR(条件1,条件2)　がTRUEやFALSEになるのは，どのようなときなのか調べてみましょう。

A列，B列にTRUE/FALSE と入力して，C列でそれらのAND演算を行い，**第5図**の表を完成させてみましょう。A1～B４のセルは，TRUEとFALSE の代わりに1と0を入力しても構いません。

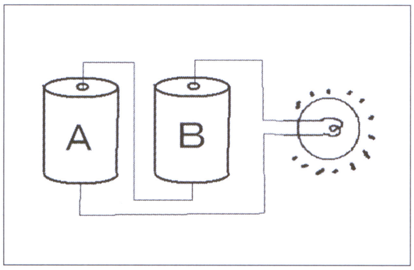
C1　fx =AND(A1,B1)

	A	B	C	D
1	TRUE	TRUE	TRUE	
2	TRUE	FALSE		
3	FALSE	TRUE		
4	FALSE	FALSE		

《第5図》エクセルでAND演算を行ってみる

《第1表》真理値表（AND）

A	B	A and B
1	1	1
1	0	0
0	1	0
0	0	0



《第6図》AND演算は直列

TRUE＝1,FALSE＝0として，AND演算の結果を**第1表**にまとめておきます。この表を真理表とか真理値表といいます。

第1表から，AND演算は積（かけ算）に似ていることに気がつくと思います。もしこの表に意味を与えるとしたら，直列に並べた乾電池と電球を考えると良いと思います。乾電池があれば１，無ければ０とします。そのとき電球が点けば１，点かなければ０とします(**第6図**)。

直列につないだ乾電池ＡとＢのどちらが欠けても電球は点きません。AND演算の真理値表は，乾電池ＡとＢが両方ともあるときだけ，電球が点くという状態を表しているのです。

同じようにしてOR演算もエクセルで行ってみましょう(**第7図**)。

OR演算の結果を真理値表にまとめると**第2表**のようになります。

OR演算は和（足し算）に似ていることに気がつくと思います。最初のOR演算は１or１＝１となり，足し算と少し違いますが，１以上はすべて１に置き換える足し算だと考えて下さい。

このOR演算の真理値表は，並列につないだ乾電池と電球の関係で意味づけることができます(**第8図**)。

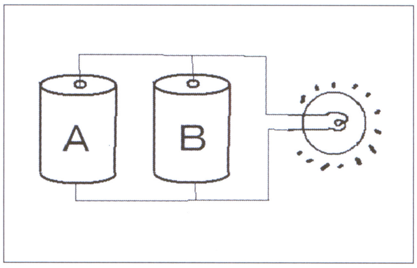
C1　fx =OR(A1,B1)

	A	B	C	D
1	TRUE	TRUE	TRUE	
2	TRUE	FALSE		
3	FALSE	TRUE		
4	FALSE	FALSE		

《第7図》エクセルでOR演算

《第2表》真理値表（OR）

A	B	A or B
1	1	1
1	0	1
0	1	1
0	0	0



《第8図》OR演算は並列

OR演算の真理値表は，乾電池ＡとＢのどちらかがあれば，電球が点くことを表しています。

【問】空欄に１または０を入れて，次の表を完成させましょう。

A	B	C	A and B and C
1	1	1	
1	1	0	
1	0	0	
0	0	0	

A	B	C	A or B or C
1	1	1	
1	1	0	
1	0	0	
0	0	0	

【解】　AND演算を積だと考えれば，答えは上から順に１,０,０,０となります。OR演算は和によく似ていることを思い出してもらえれば，答えは上から順に１,１,１,０となります。実際にANDを「＊」で，ORを「＋」で表すことはよくあります。

YAHOO!で検索

もしインターネットが使える環境ならhttp://www.yahoo.co.jpにアクセスしてみましょう。「検索オプション」をクリックすると，**第9図**のようなページが表示されます。

キーワードをいくつか並べてウェブペ

《第9図》http://www.yahoo.co.jpにアクセス

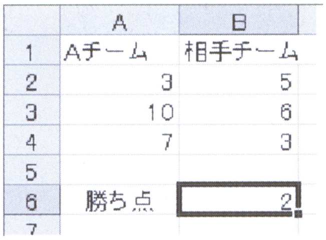
ージの検索をしてみましょう。それぞれのキーワードが存在するページは1，そうでなければ0となり，その値を足したものがOR検索，掛けたものがAND検索となります。もちろんその計算結果が1以上ならヒットしたということになります。

3番目の「エキスパート検索」も同じように考えれば，簡単に理解できます。使い方の例を見ますと，「国内（日本）のシェアウェアまたはフリーウェアを知りたい」ときは，「(シェアウェア+フリーウェア)＊(国内＋日本)」と入力すれば良いと分かります。「シェアウェア」にヒットすれば1，「フリーウェア」にヒットすれば1，などと考えて，1と0で計算させると分かりやすいですよね。

「＊」はANDで「＋」はORなんて考えるより，直接1と0の足し算と掛け算を行って結果が1以上になるかどうかと考えれば，この「エキスパート検索」も簡単に使いこなせると思います。

勝ち点の計算

相手チームに勝つと「勝ち点1」，そうでないときは「勝ち点0」が与えられる競技があるとします。3試合戦った結果，勝ち点が何点になるかエクセルで計算させてみましょう。



《第10図》エクセルで勝ち点の計算

Aチームと相手チームの試合結果は，3対5，10対6，7対3だったとします。この時，Aチームの「勝ち点2」を，エクセルでは，次のようにして計算することができます(第10図)。

セルB6に入る式は，

＝(A2＞B2)＋(A3＞B3)＋(A4＞B4)

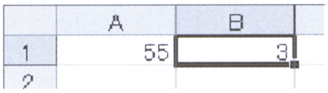
となります。

成り立てば1，そうでないときは0となる性質を利用しています。

【問】100点満点のテストを5段階評価に換算してみましょう。規則は右表になっているとします。エクセルでセルA1に点数を，セルB2に5段階に換算した結果を表示してみましょう。

80～100	5
65～79	4
45～64	3
30～44	2
0～29	1

【解】



セルB2の式は

＝(A1＞79)＋(A1＞64)＋(A1＞44)＋(A1＞29)＋1

となります。できたかな？

普通はこのような換算はVLOOKUP関数を用います。「成り立てば1」を知っていればこんな書き方もできるという例です。模範解答でありません。念のため。

電卓で確認

Windowsのメニューから「アクセサリー」「電卓」を選んで下さい(**第11図**)。電卓の種類は「関数電卓」にします(**第12図**)。

関数電卓を使うと，ANDやORの計算ができます。この電卓を使うと

0 or 1＝1　　0 and 1＝0

1 or 1＝1　　1 and 1＝1

今までやってきたことが確認できました。1以上の数字でやってみると

1 or 2＝3　　1 and 2＝0

2 or 3＝3　　2 and 3＝2

となります。おや？不思議。

この結果を理解するには，2進数の仕組みを知らなければなりません。そこで次回はちょっと地味ですが，2進数の解説を行います。

《第11図》「電卓」の画面

《第12図》「関数電卓」を選択

コラム①

今回解説したANDとORは論理演算子といいます。このほかにも，よく使う論理演算子としてXORとNOTがあります。電卓にもこのボタンがありますので，説明をしておきましょう。

XOR演算は，同じ時は0，そうでないときは1という値になります。電卓で1 Xor 1と押すと0となります。**第3表**にまとめておきます。確認して下さい。

《第3表》真理値表（XOR）

A	B	A xor B
1	1	0
1	0	1
0	1	1
0	0	0

NOT演算は，1と0を反転させます。電卓で0 Notと押して下さい。1にはなりませんが，これで良いのです。電卓では確認できませんが，**第4表**にまとめておきます。次回の説明を読めば納得してもらえると思います。

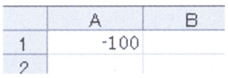
《第4表》真理値表（NOT）

A	Not A
1	0
0	1

コラム②

今回の話は，True＝1として説明してきましたが，True＝−1となることもあります。エクセルのVBAで次のプログラムを実行してみて下さい(**第13図**)。

```
Private Sub CommandButton1_Click()
Cells(1, 1).Value = True * 100
End Sub
```



《第13図》VBAの実行画面

True＊100＝−100なのでTrue＝−1とわかります。Falseは0だが，Trueは0じゃない値になると理解しておくと良いでしょう。1と−1は大した違いじゃない，なんてことも次回の説明で分かってもらえるはずです。

本コーナーは，さまざまな言語に対応したプログラミング学習と創作をテーマに，読者からの投稿作品で成り立っています。これから，プログラミングを勉強しようという皆さんや，プロのプログラマー/クリエイターを目指そうという皆さんにとって役立つ情報が満載です。手近にあるパソコンに入力して，お楽しみください。

それぞれの記事にはプログラムを入力/実行する時の注意点やプログラムの内容，改造方法などが紹介されていますので，よく読んでください。また，“CHECKER FLAG”というコラムがあり，デバッグ博士ことDr.Dがそれぞれのプログラムに対し批評を加えています。口が悪い人なので，かなりキビシイことも言っていますが，自分でプログラムを作ったり，これから投稿しようとしている人にとっては，とても参考になるでしょう。

プログラマー&クリエイターへの登竜門

投稿プログラムコーナー BM特選

プログラム投稿者へのお願い

本コーナーでは，読者の皆さんからの自作プログラム作品の投稿を受け付けています。以下の説明・注意点をよく読んで，きまりを守って投稿してください。

●受け付ける作品

プログラム作品の内容はオリジナルが基本ですが，過去に本コーナーで掲載された作品に関しては，移植版の投稿も受け付けています。また，未発表作品であることが原則ですが，投稿者自身のWebページや，投稿者が了承したWebページで掲載されているプログラムについては，投稿可とします。その際には，掲載されているWebページのURLをご記入ください。

なお，内容が他のゲームやツールに酷似している作品や，使用している素材や画像に権利的な問題があると判断した場合は，掲載を見送る場合があります。

●応募に必要なもの

①プログラム(作品)ファイル

②原稿ファイル――住所，氏名，ペンネーム(希望者)，年齢，職業もしくは学年と学校名，電話番号，eメールアドレス(お持ちの方)を，必ずお書きください。

プログラムの説明には，使用言語名／作品タイトル／実行方法／作品の内容／操作方法／プログラムの入力方法／改造のアドバイス／制作に苦労した点／参考にしたもの／などをお書きください。

●応募方法

郵送もしくはeメールにて受け付けます。

①郵送による投稿

用意したプログラムファイルと原稿ファイルを，フロッピーディスクもしくはCD-Rにセーブし，メディアのラベルなどに，作品タイトルと使用言語名，氏名を記述してください。郵送する封筒の裏面にも，住所，氏名，作品タイトル，使用言語名を記述してください。また本誌とじ込みの愛読者カードについている応募カードも忘れずに同封してください。

あて先：〒141-8715　東京都品川区東五反田1-11-15　電波新聞社マイコンBASICマガジン編集部「投稿プログラムコーナー」係

② eメールによる投稿

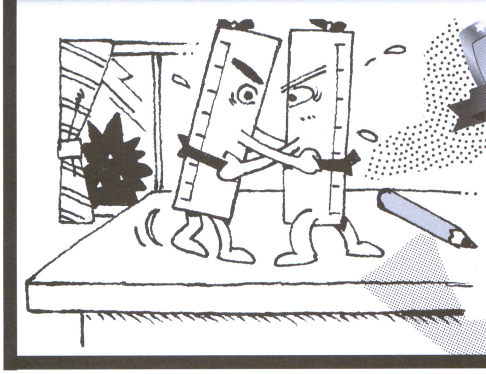
用意したプログラムファイルと原稿ファイルを，LHAなどで圧縮して一つのファイルにし，そのファイルをメールに添付のうえ，下記アドレスまで送信してください。その際，圧縮ファイルの添付し忘れにご注意ください。

アドレス：webmas@basicmagazine.net

件名：BM投稿プログラム

なお，掲載されたプログラムに関しては，規定の原稿料（1万円ー税金＝9千円）をお支払いし，掲載誌をお送りいたします。

Visual Basic V6.0用



定規同士の戦い
ジョーピン

大石　晋也　(21歳)

内容

机の上に定規を置き，その上からペンの後ろ側を押し当て，そのまま定規の端まで動かすと，ガクッという音とともに定規が動きます。こうやって定規を操り，相手の定規を机から落とす遊び「ジョーピン」が，わが小中学生時代に一世を風びしました (他の地域にも，こういった遊びがあるかは不明)。

この「ジョーピン」をVBで再現しました。以前にも投稿したのですが，2002年10月号でもう一歩だったので，その作品を改良しました。主に，ルールや操作性の向上を図りました。

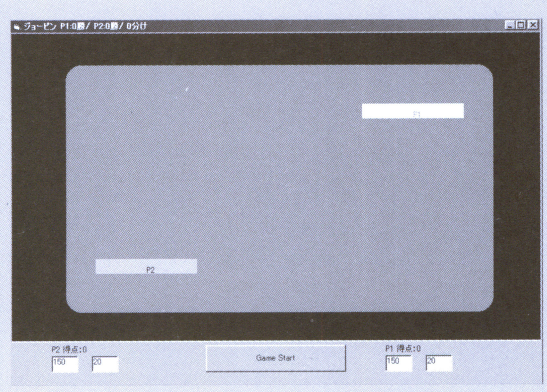
遊び方

2人対戦用です。画面右上の白い定規がプレイヤー1，左下の緑の定規がプレイヤー2です(写真1)。

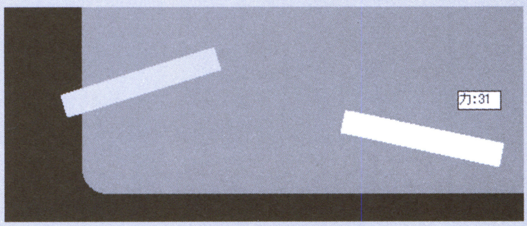
まず，下のほうにあるテキストボックスに数値を入力して，定規の大きさを決めてください。画面下の中央にあるGame Startボタンを押すとゲーム開始です。

定規の中心に表示される文字が赤くなっている方が定規を動かすプレイヤーです。また，現在の状況が下のピクチャボックスに表示されます。

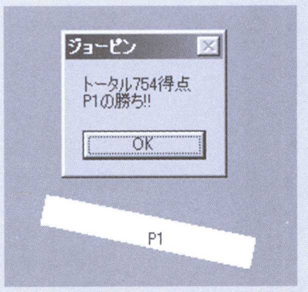
定規の動かし方は，まず自分の定規の上にマウスのカーソルをもっていき，マ



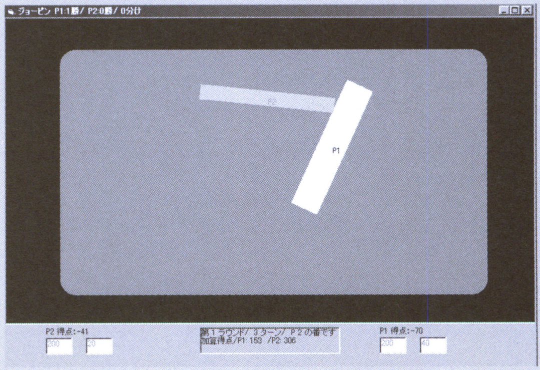
《写真1》定規同士でぶつけ合う対戦ゲーム



《写真2》力をためて相手の定規にぶつけろ！



《写真3》総合得点で勝敗が決まる



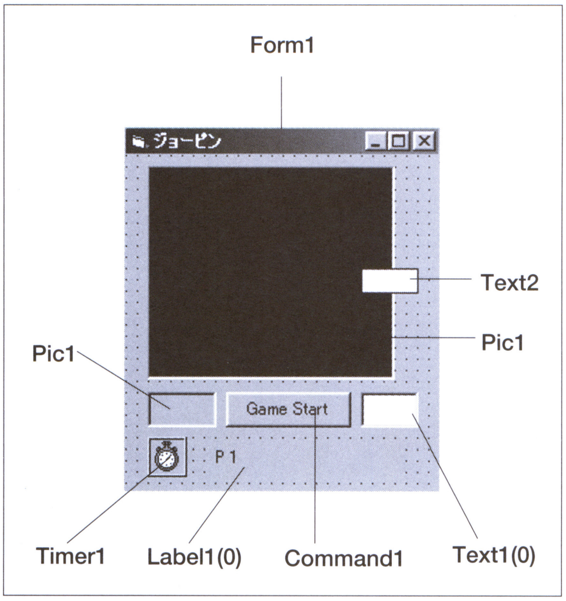
《写真4》さまざまな大きさの定規に設定できる

《OVER FLOW》2003年新春・2月号です。創刊250号を目の前にし，ますますがんばるベーマガ（本誌の愛称）です。新連載なども入り始めましたが，いかがでしょう。読者のためのベーマガです。みなさんの意見が本誌を変えていきます。お便りよろしく！

ウスのボタンを押します。すると，定規の中心あたりに現在ためている力が表示され，その数がどんどん増えていきます(**写真２**)。これが定規を打ち出すときの力です。大きいほど速く，遠くまで滑ります。

ある程度力がたまったら(デフォルトで15程度，定規が大きいほど余計に力が要ります)，マウスのボタンを押すのをやめ，定規の端へカーソルをじっくりもっていくと，定規が飛んでいきます。

このとき，あまり速く動かすと，スカしてしまいます。この場合，もう一度力をため直してください。力をためすぎたなと思ったときも，再度クリックすることで，再び力をため直すことができます。また，定規が机の外へはみ出している部分からは打てません。



《第1図》フォームの配置図

実際の定規を動かすときと同じ原理を再現しているので，定規は動かしたカーソルとは逆方向に滑っていきます。

あまり細長い定規で力を込めて打つと，プログラムの仕組み上，定規同士がかさなってしまうことがあるので，ほどほどに。

ルール

１試合４ラウンドの得点制です。各ラウンドごとの勝者に得点がプラスされていきます。定規が小さいほど加算得点は大きくなります。チョット特殊なルールとして，１ターン未満で相手を落とすと，落とされた相手に得点が加算されます。

また，定規を動かすたびに，加えた力の分だけ得点から引かれていきます。ターンが大きくなるほど勝者に加えられる得点は大きくなります。

１ラウンドごとに定規の大きさを変えられるので，現在の得点具合から戦略を考えてください。

苦労した点

リストのシェイプアップ。機械的であまり見る意味がないところは，どんどん簡略化したせいか，ぱっと見ただけでは，なにを意味しているのかわかりにくいです。地味な画面のくせに**リスト**が少し長めです。これでもだいぶ減らしたのですが…。

また，定規間の摩擦を計算していないので，完全な「ジョーピン」の再現にはなっていません。

★参考にしたもの

・工業力学(入江敏博/山田 元)

・マイコンBASICマガジン2002年10月号105ページ影さんのコメント

CHECKER FLAG

影：2002年10月号の「もう一歩」のときにもコメントしたけど，ゲーム設定のアイデアはバッチリ。また，今回のバージョンで定規の初期位置や操作方法などが修正されたことで，ゲームとしても遊びやすくなった印象。

Dr.D：じゃが，やはり２人対戦専用はいただけない。２人いるのだったら，実際の定規で「ジョーピン」をした方が楽しいからのぉ。こういったゲームは，COM戦がないとゲーム化した意味が半減してしまうゾ。

つぐ美：こういった定規を使った遊びって小学生時代に誰でもやっているんじゃないかしら？

編：ペンでなく消しゴムを使うと，定規がよく飛ぶんですよ。また，相手の定規の上に乗せるという技もありましたね。

《第１表》プロパティ表

フォームの設定	
Name	Form1
Caption	ジョーピン
Height	3960
Left	0
LinkTopic	Form1
ScaleHeight	237
ScaleMode	3 'ﾋﾟｸｾﾙ
ScaleWidth	223
StartUpPosition	3 'Windows の既定値
Top	0
Width	3465
ピクチャボックスの設定	
Name	Pic1
AutoRedraw	-1 'True
BackColor	&H00000000&
Height	2265
Left	240
ScaleHeight	147
ScaleMode	3 'ﾋﾟｸｾﾙ
ScaleWidth	172
TabIndex	0
Top	120
Width	2640
Name	Pic2
Height	375
Left	240
ScaleHeight	315
ScaleWidth	675
TabIndex	5
Top	2520
Width	735
テキストボックスの設定	
Name	Text1(0)
Height	375
Index	0
Left	2520
TabIndex	4
Top	2520
Width	615
Name	Text2
Appearance	0 'ﾌﾗｯﾄ
Height	270
Left	2520
TabIndex	1
Top	1200
Width	615
コマンドボタンの設定	
Name	Command1
Caption	Game Start
Height	375
Left	1080
TabIndex	3
Top	2520
Width	1335
ラベルの設定	
Name	Label1(0)
Caption	P 1
Height	255
Index	0
Left	960
TabIndex	2
Top	3120
Width	1980
タイマーの設定	
Name	Timer1
Interval	150
Left	240
Top	3000

《リスト》ジョーピンのプログラム・リスト

```
Private Declare Function RoundRect Lib "gdi32" (ByVal hdc As Long, _
        ByVal X1 As Long, ByVal Y1 As Long, ByVal X2 As Long, ByVal Y2 As Long, _
        ByVal ellipseWidth As Long, ByVal ellipseHeight As Long) As Long
Private Declare Function Polygon Lib "gdi32" (ByVal hdc As Long, _
        lpPoint As POINTAPI, ByVal nCount As Long) As Long
Private Declare Function GetTickCount Lib "kernel32" () As Long
Private Declare Function MoveWindow Lib "User32" (ByVal hWnd As Long, _
        ByVal X As Long, ByVal Y As Long, ByVal nWidth As Long, _
        ByVal nHeight As Long, ByVal bRepaint As Long) As Long
Private Declare Function GetSystemMetrics Lib "User32" (ByVal nIndex As Long) As Long

Private Type POINTAPI
    X As Long
    Y As Long
End Type

Const Rd = 3.141592 / 180

Dim Term As Single 'ターン数
Dim Act As Integer '攻撃者
Dim Win As Integer '勝利者
Dim Lp As Integer 'ループ中か
Dim PN  As Integer 'プレイヤーの番号
Dim Round As Byte 'ラウンド
Dim Wait As Long  'ウエイト

'以下の配列の"1 to 2"はプレイヤー1,2のことです
Dim S(1 To 2) As Single '定規の面積かつ質量
Dim Pt(4, 1 To 2) As POINTAPI '定規の四方の位置.0=4
Dim Pt2(4, 1, 1 To 2) As Single 'Ptの正確な位置.4-4つの点/1-XとY座標
Dim DPt2(4, 1, 1 To 2) As Single '位置を一時保存する
Dim Gx(1 To 2) As Long, Gy(1 To 2) As Long '定規の重心
Dim V(2, 1 To 2, 0 To 2) As Single '定規の速度/2はプレイヤー1のダミー/縦横回転の速度
Dim Rx(1 To 2) As Integer, Ry(1 To 2) As Integer '重心から力の働く点までの距離
Dim Mt(1 To 2) As Single '定規のモーメント
Dim P(1 To 2) As Byte '定規を飛ばすときに横切った線の番号
Dim VtX(1 To 2) As Double, VtY(1 To 2) As Double '定規を飛ばすときのベクトル
Dim Vt(1 To 2) As Single 'ベクトルの大きさ
Dim Out(1 To 2) As Byte '場外に出たか
Dim WinP(0 To 3) As Integer '勝率,0は引き分け
Dim T(1 To 2) As Single '得点
Dim Jc(1 To 2) As Long '定規の色

Dim Dx As Integer, Dy As Integer 'デスクトップの領域
Dim Power As Single  '力
Dim Start As Byte '力をため始めたか
Dim Flag As Byte '定規の衝突判定を交互に行なうためのフラグ
Const A = 0.01, K = 0.5 'A-回転速度の係数/K-摩擦によって失われる速度の係数
Dim Size(3) As Single '定規四辺の半分の大きさ
Dim L As Integer, J As Byte, I As Byte, M As Byte, N As Byte, H As Byte  '雑用
```

```
Private Sub Form_Load()
    Timer1.Interval = 150
    Form1.AutoRedraw = True
    Form1.ScaleMode = vbPixels
    Dx = GetSystemMetrics(16): Dy = GetSystemMetrics(17)
    Call MoveWindow(Form1.hWnd, 0, 0, Dx, (Dy + GetSystemMetrics(4)), 1)
    Call MoveWindow(Pic1.hWnd, 0, 0, Dx, Dy - 70, 1)
    Pic1.BackColor = vbBlack
    With Text1(0): .Width = 40
                   .Height = 25
                   .Top = Pic1.Height + 20
                   .Left = Pic1.Width * 0.7
                   .Text = 150: End With
    For L = 1 To 3
        Load Text1(L)
        With Text1(L): .Visible = True
                       .Width = 40
                       .Height = 25
                       .Top = Text1(0).Top
                       .Left = Text1(L - 1).Left * Abs(L - 2) + 60
                       .Text = 150 - 130 * Abs(L - 2): End With
    Next L
    Load Label1(1)
    For L = 0 To 1
        With Label1(L): .Visible = True
                        .Width = 100
                        .Height = 17
                        .Top = Pic1.Height + 5
                        .Left = Text1(L * 2).Left
                        .Caption = "P " & L + 1: End With
    Next L
    Command1.Move (Form1.ScaleWidth - 210) / 2, Pic1.Height + 5, 210, 40
    Round = 1
    Act = 1
    P(1) = 5: P(2) = 5
    Jc(1) = vbWhite: Jc(2) = vbGreen
    Call Init
End Sub
'初期化
Private Sub Init()
    Term = 0: Win = -1: Lp = 0
    Form1.Caption = "ジョーピン P1:" & WinP(1) & "勝/ P2:" & WinP(2) & "勝/ " & WinP(3) & "分
け"
    For L = 0 To 1
        Label1(L).Caption = "P" & L + 1 & " 得点：" & T(L + 1)
        Out(L + 1) = 0
        Gx(L + 1) = Pic1.Width * (3 - 2 * L) / 4
        Gy(L + 1) = Pic1.Height * (1 + 2 * L) / 4
        For M = 1 To 2
            For N = 0 To 2
                V(L, M, N) = 0
            Next N
        Next M
```

★去年はあまりゲームを作っていない気がする。これではいかんな，プチゲーム職人（自称）として！（青森県南津軽郡・ていこく23歳）……【**Dr.D：**それはいかんなー。デバッグ博士としては，投稿プログラムを批評するのが楽しみなんじゃがな。よろしく！】

```
    Next L
    Pic1.FillStyle = vbSolid
    Call DrawDesk
    Call SetSize
    Text2.Visible = False
    Pic2.Visible = False
    Command1.Enabled = True
    Command1.Visible = True
    For L = 0 To 3
        Text1(L).Enabled = True
    Next L
End Sub

Private Sub Text1_Change(Index As Integer)
    On Error GoTo EX
    If Text1(Index).Text > 999 Then Text1(Index).Text = 20
    Call DrawDesk
    Call SetSize
EX:
End Sub
'机を描く
Private Sub DrawDesk()
    Call SetPen(1, RGB(180, 130, 80), RGB(180, 130, 80))
    Pic1.Cls
    Call RoundRect(Pic1.hdc, 0.1 * Pic1.Width, 0.1 * Pic1.Height, 0.9 * Pic1.Width, 0.9 *
Pic1.Height, 0.1 * Pic1.Height, 0.1 * Pic1.Height)
End Sub
'描画を設定
Private Sub SetPen(W As Integer, FillC As Long, ForeC As Long)
    With Pic1: .DrawWidth = W
               .FillColor = FillC
               .ForeColor = ForeC: End With
End Sub
'定規の大きさ
Private Sub SetSize()
    For M = 0 To 3
        Size(M) = Val(Text1(M).Text) / 2
    Next M
    For M = 1 To 2
        For L = 0 To 4
            Pt(L, M).X = Gx(M) + Sgn(L ^ 2 - 5 * L + 5) * Size(2 * (M - 1))
            Pt(L, M).Y = Gy(M) + Sgn(4 * L ^ 2 - 12 * L + 7) * Size(2 * M - 1)
            Pt2(L, 0, M) = Pt(L, M).X
            Pt2(L, 1, M) = Pt(L, M).Y
        Next L
        S(M) = (Size(M * 2 - 2) * 2) * (Size(M * 2 - 1) * 2) / 3500
        Call SetPen(2, Jc(M), Jc(M))
        Call Polygon(Pic1.hdc, Pt(1, M), 4)
        Call SetPen(1, vbBlack, vbBlack)
        If Act = M Then Call SetPen(1, vbRed, vbRed)
        Pic1.CurrentX = Gx(M): Pic1.CurrentY = Gy(M)
        Pic1.Print "P" & M
    Next M
End Sub

Private Sub Command1_Click()
    Call DrawDesk
    Call SetSize
    For L = 0 To 3
        If Size(L) < 8 Then MsgBox "小さすぎます。": Exit Sub
    Next L
    If S(1) > 2.5 Or S(2) > 2.5 Then MsgBox "大きすぎます。": Exit Sub
    For L = 0 To 3
        Text1(L).Enabled = False
    Next L
    Command1.Visible = False
    Pic2.Move Command1.Left, Command1.Top, Command1.Width, Command1.Height
    Pic2.Visible = True
End Sub

Private Sub Pic1_MouseDown(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Start = 0 Then Power = 0
    If Lp <> 0 Then Exit Sub
    With Text2: .Move Gx(Act), Gy(Act)
                .Visible = True
                .Text = "力：" & Power: End With
    For M = 1 To 2
        Call SetPen(2, Jc(M), Jc(M))
        Call Polygon(Pic1.hdc, Pt(1, M), 4)
        If Act = M And Pic1.Point(X, Y) = Jc(M) Then Start = 1: PN = M
    Next M
End Sub

Private Sub Pic1_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Power > 0 Then Call Judge(X, Y)
End Sub
'カーソルが定規の端にあるか
Private Sub Judge(X As Single, Y As Single)
    For L = 0 To 3
        Call DrawDesk
        Call SetPen(9, vbBlack, vbRed)
        If Pic1.Point(X, Y) = vbBlack Then '机の外では撃てない
            PN = Abs(PN - 2) + 1
            Exit For
        End If
        Pic1.Line (Pt2(L, 0, PN), Pt2(L, 1, PN))-(Pt2(L+1, 0, PN), Pt2(L+1, 1, PN))
        Call SetPen(1, Jc(PN), Jc(PN))
        Call Polygon(Pic1.hdc, Pt(1, PN), 4)
        If Pic1.Point(X, Y) = vbRed Then
            T(PN) = T(PN) - Power
            Label1(PN - 1).Caption = "P" & PN & " 得点：" & T(PN)
            P(PN) = L
            If P(PN) = 0 Then P(PN) = 4 '以下初速度を計算
            VtX(PN) = Pt2(P(PN) - 1, 0, PN) - Pt2(P(PN), 0, PN)
            VtY(PN) = Pt2(P(PN) - 1, 1, PN) - Pt2(P(PN), 1, PN)
```

★PC-Engineのでべろ，PC-FXGA，GAME BASIC for SEGASATURNも過去載せてたんですか？僕はこの三つがどうしてもほしくて，プログラムもしてみたい。情報をください。お願いします。（茨城県稲敷郡・**雪崩式21歳**）……**【影：ベーマガの掲示板で読者に尋ねてみては？】**

```
            Vt(PN) = Sqr(VtX(PN) ^ 2 + VtY(PN) ^ 2)
            V(0, PN, 0) = Power * (VtX(PN) / Vt(PN)) / S(PN)
            V(0, PN, 1) = Power * (VtY(PN) / Vt(PN)) / S(PN)
            V(1, PN, 0) = V(0, PN, 0):  V(1, PN, 1) = V(0, PN, 1)
            Rx(PN) = (X - Gx(PN)):      Ry(PN) = (Y - Gy(PN))
            Mt(PN) = Power * (Rx(PN) * VtY(PN) - Ry(PN) * VtX(PN)) / Vt(PN)
            V(0, PN, 2) = A * Mt(PN)
            V(1, PN, 2) = V(0, PN, 2) 'ここまで
            Power = 0: Start = 0
            Call AttackLoop
            Exit Sub
        End If
    Next L
    Call DrawDesk
    For L = 1 To 2
        Call SetPen(2, Jc(L), Jc(L))
        Call Polygon(Pic1.hdc, Pt(1, L), 4)
        If PN = L And Pic1.Point(X, Y) <> Jc(L) Then
            Start = 0: Power = 0
            Text2.Text = "スカ"
        End If
    Next L
End Sub

Private Sub Pic1_MouseUp(Button As Integer,Shift As Integer,X As Single,Y As Single)
    For L = 1 To 2
        Call SetPen(1, vbBlack, vbBlack)
        If Act = L Then Call SetPen(1, vbRed, vbRed)
        Pic1.CurrentX = Gx(L): Pic1.CurrentY = Gy(L)
        Pic1.Print "P" & L
    Next L
    Start = 0
End Sub

Private Sub Text2_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
    Text2.Top = Text2.Top - 50
End Sub

Private Sub Timer1_Timer()
    If Command1.Visible = True Then Exit Sub
    If Start > 0 Then
        Power = Power + 1
        Text2.Text = "力：" & Power
    End If
    If Power = 0 Then
        Pic2.Cls: Pic2.CurrentX = 0: Pic2.CurrentY = 0
        Pic2.Print "第"; Round; "ラウンド/ "; Term; "ターン/  "; "P"; Act; "の番です"
        Pic2.Print "加算得点/"; "P1:"; Fix((Term + 0.5) * 100 / S(1)); " /P2:"; Fix((Term +
0.5) * 100 / S(2))
    End If
End Sub
'次の位置を計算
Private Sub MoveJogi(PN2 As Byte, Theta As Single, MoveX As Single, MoveY As Single)
    Dim X, Y, NewX As Single
    For N = 0 To 4
        For H = 1 To 2
            DPt2(N, 0, H) = Pt2(N, 0, H)
            DPt2(N, 1, H) = Pt2(N, 1, H)
        Next H
        X = Pt2(N, 0, PN2) - Gx(PN2)
        Y = Pt2(N, 1, PN2) - Gy(PN2)
        NewX = X * Cos(Theta) - Y * Sin(Theta)  '─┐
        Y = X * Sin(Theta) + Y * Cos(Theta)     '─┴回転
        Pt2(N, 0, PN2) = NewX + Gx(PN2) + MoveX '─┐
        Pt2(N, 1, PN2) = Y + Gy(PN2) + MoveY    '─┴平行移動
        Pt(N, PN2).X = Pt2(N, 0, PN2)
        Pt(N, PN2).Y = Pt2(N, 1, PN2)
    Next N
End Sub
'攻撃のループ処理
Private Sub AttackLoop()
    Lp = 10
    While Lp > 0
        Call Attack
    Wend
    For L = 1 To 2
        If Out(L) = 1 And Term >= 1 Then Win = Abs(L - 2) + 1
        If Out(L) = 1 And Term < 1 And Act = Abs(L - 2) + 1 Then Win = L
        If Out(L) = 1 And Term < 1 And Act = L Then Win = Abs(L - 2) + 1 '自滅防止
    Next L
    If Out(1) = 1 And Out(2) = 1 And Term >= 1 Then Win = 0
    Term = Term + 0.5
    Act = Abs(Act - 2) + 1 '1と2を入れ替え
    Text2.Visible = False
    For L = 1 To 2
        Call SetPen(1, vbBlack, vbBlack)
        If Act = L Then Call SetPen(1, vbRed, vbRed)
        Pic1.CurrentX = Gx(L)
        Pic1.CurrentY = Gy(L)
        Pic1.Print "P" & L
    Next L
    If Win = -1 Then Exit Sub
    If Win = 1 Or Win = 2 Then
        MsgBox "第" & Round & "ラウンド P" & Win & "の勝利!!  " & Chr(13) & Fix(Term * 100 /
S(Win)) & "ポイント獲得"
        T(Win) = T(Win) + Fix(Term * 100 / S(Win))
        Label1(Win - 1).Caption = "P" & Win & " 得点：" & T(Win)
        Round = Round + 1
        If Round = 5 Then
            Win = (1.5 * Abs(Sgn(T(2) - T(1)))) + Sgn(T(2) - T(1)) / 2
            If Win = 1 Or Win = 2 Then
                MsgBox "トータル" & T(Win) & "得点 " & Chr(13) & "P" & Win & "の勝ち!!"
                WinP(Win) = WinP(Win) + 1
            ElseIf Win = 0 Then
                MsgBox "引き分け"
                WinP(0) = WinP(0) + 1
```

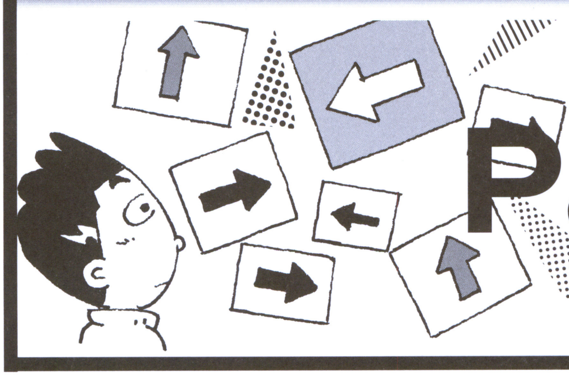
★ジョジョダジャレ　“ジョセフが饒舌にしゃべった”“ディオがデオドラントをつけた”。(福岡市・**前田宣弘28歳**)……【**つぐ美：**ダジャレ評論家の影さん，いかがでしょう。**影：**うーん，饒舌って漢字が読めないとダジャレにもならんですな，こりゃ。私は読めるさ。】

```
            End If
            Round = 1
            T(1) = 0: T(2) = 0
        End If
    ElseIf Win = 0 Then
        MsgBox "引き分け"
        WinP(0) = WinP(0) + 1
    End If
    If Win <> -1 Then Call Init
End Sub
'攻撃
Private Sub Attack()
    For M = 1 To 2
        Call MoveJogi((M), (V(1, M, 2) * Rd), (V(1, M, 0)), (V(1, M, 1)))
        Gx(M) = (Pt2(1, 0, M) + Pt2(2, 0, M) + Pt2(3, 0, M) + Pt2(4, 0, M)) / 4
        Gy(M) = (Pt2(1, 1, M) + Pt2(2, 1, M) + Pt2(3, 1, M) + Pt2(4, 1, M)) / 4
        Flag = Flag + 1
        Call Collision
    Next M
    Flag = 0
    For M = 0 To 4
        For N = 1 To 2
            DPt2(M, 0, N) = Pt2(M, 0, N)
            DPt2(M, 1, N) = Pt2(M, 1, N)
        Next N
    Next M
    Call DrawDesk
    For M = 1 To 2 '場外かの判定
        Out(M) = 1 'まず場外にいることにする
        If Gx(M) > 0.1 * Pic1.Width And Gx(M) < 0.9 * Pic1.Width And _
            Gy(M) > 0.1 * Pic1.Height And Gy(M) < 0.9 * Pic1.Height Then Out(M) = 0
        If Pic1.Point(Gx(M), Gy(M)) = vbBlack Then Out(M) = 1 '机の角を判定
    Next M
    For M = 1 To 2
        Call SetPen(2, Jc(M), Jc(M))
        Call Polygon(Pic1.hdc, Pt(1, M), 4)
    Next M
    Do
        DoEvents
    Loop Until GetTickCount - Wait >= 40 '<--ウエイト
    Wait = GetTickCount
    For M = 1 To 2
        For N = 0 To 2
            If V(1, M, N) <> 0 Then V(1, M, N) = V(1, M, N) - V(1, M, N) / Abs(V(1, M, N)) * K
* S(M)
            If Abs(V(0, M, N)) < Abs(V(1, M, N)) Then V(1, M, N) = 0
            V(0, M, N) = V(1, M, N)
        Next N
    Next M
    If Out(1) = 1 And Out(2) = 1 Then Lp = Lp - 1
    If V(1, 1, 0)=0 And V(1, 1, 1)=0 And V(1, 1, 2)=0 And Out(2)=1 Then Lp = Lp - 1
    If V(1, 2, 0)=0 And V(1, 2, 1)=0 And V(1, 2, 2)=0 And Out(1)=1 Then Lp = Lp - 1
    If V(1, 1, 0)=0 And V(1, 1, 1)=0 And V(1, 1, 2)=0 And _
```

```
        V(1, 2, 0) = 0 And V(1, 2, 1) = 0 And V(1, 2, 2) = 0 Then Lp = 0
End Sub
'衝突の判定
Private Sub Collision()
    For N = 1 To 4
        For L = 2 To 1 Step -1
            Call SetPen(1, Jc(L), Jc(L))
            Call Polygon(Pic1.hdc, Pt(1, L), 4)
            If Pic1.Point(Pt(N, Abs(L-2)+1).X, Pt(N, Abs(L-2)+1).Y) = Jc(L) Then
                For H = 1 To 2
                    For I = 0 To 4
                        Pt2(I, 0, H) = DPt2(I, 0, H)
                        Pt2(I, 1, H) = DPt2(I, 1, H)
                        Pt(I, H).X = Pt2(I, 0, H)
                        Pt(I, H).Y = Pt2(I, 1, H)
                    Next I
                    Gx(H) = (Pt2(1,0,H) + Pt2(2,0,H) + Pt2(3,0,H) + Pt2(4,0,H)) / 4
                    Gy(H) = (Pt2(1,1,H) + Pt2(2,1,H) + Pt2(3,1,H) + Pt2(4,1,H)) / 4
                Next H
                DoEvents '衝突を描画
                Call ChangeV((N))
                If Flag = 2 Then Call Attack
                Exit Sub
            End If
        Next L
    Next N
End Sub
'衝突による速度の変化
Private Sub ChangeV(H As Byte)
    V(2, 1, 0) = ((S(1) - S(2)) * V(1, 1, 0) + 2 * S(2) * V(1, 2, 0)) / (S(1) + S(2))
    V(2, 1, 1) = ((S(1) - S(2)) * V(1, 1, 1) + 2 * S(2) * V(1, 2, 1)) / (S(1) + S(2))
    Rx(1) = (Pt2(H, 0, 2) - Gx(1))
    Ry(1) = (Pt2(H, 1, 2) - Gy(1))
    Mt(1) = S(2) * (Rx(1) * V(2, 1, 1) - Ry(1) * V(2, 1, 0))
    V(1, 1, 2) = A * Mt(1)
    V(1, 2, 0) = ((S(2) - S(1)) * V(1, 2, 0) + 2 * S(1) * V(1, 1, 0)) / (S(1) + S(2))
    V(1, 2, 1) = ((S(2) - S(1)) * V(1, 2, 1) + 2 * S(1) * V(1, 1, 1)) / (S(1) + S(2))
    Rx(2) = (Pt2(H, 0, 1) - Gx(2))
    Ry(2) = (Pt2(H, 1, 1) - Gy(2))
    Mt(2) = S(1) * (Rx(2) * V(1, 2, 1) - Ry(2) * V(1, 2, 0))
    V(1, 2, 2) = A * Mt(2)
    V(1, 1, 0) = V(2, 1, 0): V(1, 1, 1) = V(2, 1, 1)
    For J = 1 To 2
        V(0, J, 0) = V(1, J, 0)
        V(0, J, 1) = V(1, J, 1)
        V(0, J, 2) = V(1, J, 2)
    Next J
End Sub
```

★寒いですね。おでん缶とワンカップで飲んでいるとわびしくて，ナイスです。（愛知県知多郡・**四谷一獲28歳**）……【**影：**編集長は最近，店に飲みに行く回数が減ったようですが，最寄駅から自宅までワンカップが手放せなくなったそうです。**編集長：**なぜそれを！】

Visual Basic V6.0用



矢印パネルを消去

Panel Room（パネル ルーム）

井上　秀夫 (26歳)

内容

矢印パネルを用いたパズルゲームです。矢印パネルを移動させたり，トラップパネルを利用したりして，画面上のすべての矢印パネルを消去するとステージクリアとなります。

遊び方

タイトル画面(**写真1**)に表示されている「⬅➡」をクリックしてステージを選択します(ステージは11まで)。ステージ選択後，STARTをクリックするとゲームスタートです。

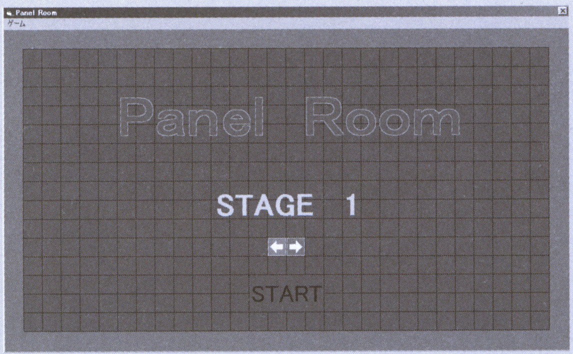
ゲームスタート後，画面上(**写真2**)に表示される矢印パネルをクリックすると，矢印の方向にパネルは1マス進みます。同じ色のパネル同士をくっつけると，そのくっつけた2枚の矢印パネルが消去されます。

そして矢印パネルとは別に，3種類のトラップパネル(**写真3**)が存在します。矢印パネルが紫色のトラップパネルの上を通過すると，矢印の向きが変わります。トラップパネルの詳しい効果は，画面右に表示されています。ステージ1以外は，このトラップパネルを利用しないとクリアできません。

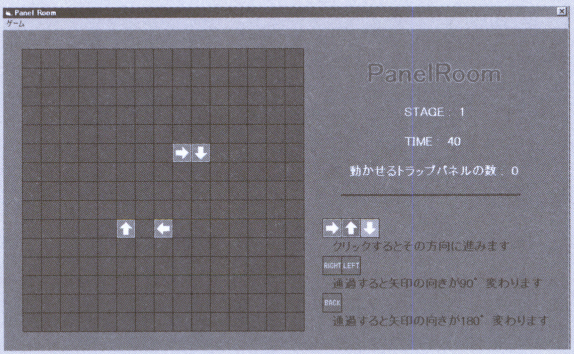
また，トラップパネルは，決められた回数(画面右に表示されている「動かせるトラップパネルの数」)以内なら移動させることもできます。パネルの上でボタンを押すとポインタが十字になるので，目的の方向にポインタを1マス移動し，ボタンを離します(そう，デスクトップでアイコンを移動する時のように)。

プレイ中，失敗した場合は，メニューバーのゲームにあるコマンドで，一手戻したり，最始からやり直すことができます。

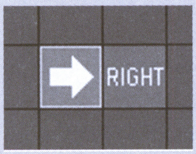
タイムオーバー(制限時間は，画面右のTIMEに表示されています)に注意して，全ステージクリアを目指してくださ



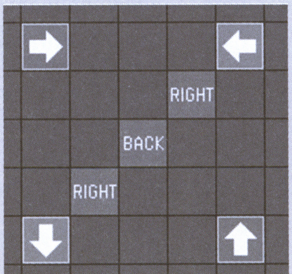
《写真1》タイトル画面でステージセレクトも可能



《写真2》同じ色の矢印同士を重ねる

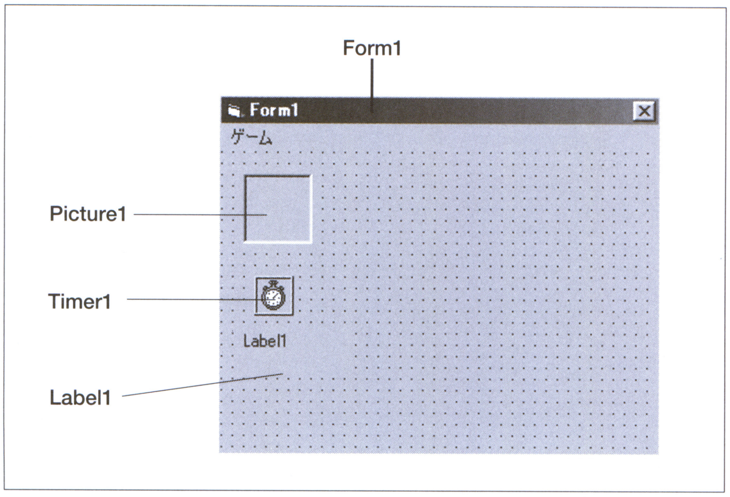


《写真3》このトラップパネルが重要



《写真4》思考力が必要となるステージ構成

★ネタがない，あーネタがない，ネタがない。これって過去にも載ったようなネタですか？……あれ？これってネタだよね。ネタあったんじゃー。(石川県小松市・**プロロ26歳**)……【**影：**ネタ，ネタって，あんたは寿司屋か！　うん？私は仕事中に寝てない！】



《第1図》フォームの配置図

い。最終ステージをクリアすると，偉人からのメッセージを見ることができます。

入力について

第1図を参考に，フォームにピクチャーボックス，タイマ，ラベルを一つずつ貼り付け，**第１表**を参考にプロパティを設定してください。プロパティのほとんどは，コード内で設定していますが，Form1のBorderStyleのプロパティは，プロパティウィンドウで設定してください。

第２表を参考に，メニューエディタでメニューも作ります。

効果音のWavファイル，画像ファイルの**panel.bmp**も作成してください。

あとは，**リスト**を入力します。注意として，**リスト３**(data.txt)の末尾の３行は，空白も入れて必ず各20文字になるようにしてください。

《第１表》プロパティ表

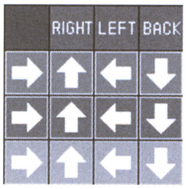
フォームの設定	
Name	Form1
BorderStyle	1 '固定(実線)
Caption	Form1
ForeColor	&H00000000&
Height	3855
Left	105
LinkTopic	Form1
MaxButton	0 'False
MinButton	0 'False
ScaleHeight	213
ScaleMode	3 'ﾋﾟｸｾﾙ
ScaleWidth	312
StartUpPosition	3 'Windows の既定値
Top	105
Width	4770
ピクチャボックスの設定	
Name	Picture1
Height	735
Left	240
ScaleHeight	675
ScaleWidth	675
TabIndex	0
Top	240
Width	735
ラベルの設定	
Name	Label1
Caption	Label1
Height	375
Left	240
TabIndex	1
Top	1920
Width	1095
タイマーの設定	
Name	Timer1
Left	360
Top	1320

改造

ステージを追加するときは，まずコード冒頭のs＄()の括弧の中の値を，追加したい分だけ増やし，Form_Load()内の変数stmの値を，それと同じ数にしてください。

次にdata.txt(**リスト３**)を編集します。上から11行にわたって，数字とアルファベットが羅列しています。これがステージデータです。

例えばステージ12を作る場合は，上から12行目に新たに行を設け，最初に作成



panel.bmp

《第３表》変数表

s＄()	ステージデータ
px,py	キャラクタを表示する座標
c	表示するキャラクタの種類
xd,yd	矢印が進む方向
p	矢印パネルの総数
k	動かせるトラップパネルの数
st	ステージ数
tm	残りタイム
tit	タイトル画面であるかどうか
stm	総ステージ数
ch	トラップパネルを通過したかどうか
ct	進んだ手数
tr	マウスポインタが十字であるかどうか
mm,mxy	押されたパネルの種類
md	押された矢印が進む先の状態
mx,my	マウスボタンが押された時のパネルの座標
lx,ly	マウスボタンが離された時のパネルの座標
t1,t2	wait関数の作成で使用
e＄(3)	ステージデータ(パネルの座標，種類)
b(500,15,15)	ゲームの過程(手数と15×15マスの内容)
kk(500)	一手一手その時の動かせるトラップパネルの数
pp(500)	一手一手その時の矢印パネルの総数
ms＄(3)	メッセージのデータ
m(17,17)	仮想画面
m1,m2	文字の表示位置
m3	表示する文字
m4	文字の色
m5	文字のサイズ
z	効果音を鳴らす
ea,eb,ec,ed,ow,wt＄,ag＄,aa＄()	エンディングで使用

《第４表》パネルの種類と対応する文字

文字	パネルの種類
0	普通のマス
1	RIGHT
2	LEFT
3	BACK
4	青の→
5	青の↑
6	青の←
7	青の↓
8	緑の→
9	緑の↑
a	緑の←
b	緑の↓
c	赤の→
d	赤の↑
e	赤の←
f	赤の↓

《第２表》メニューの設定

ｷｬﾌﾟｼｮﾝ(P)	名前(M)	ｲﾝﾃﾞｯｸｽ(X)	ﾍﾙﾌﾟｺﾝﾃｷｽﾄ番号(H)	ｼｮｰﾄｶｯﾄ(S)	ﾈｺﾞｼｴｰﾄ位置(O)	ﾁｪｯｸ(C)	有効(E)	表示(V)	ｳｲﾝﾄﾞｳﾘｽﾄ(W)
ゲーム	game		0	(なし)	0 ーなし		○	○	
…一手戻す	modosu		0	(なし)	0 ーなし		○	○	
…やり直し	yarinaosi		0	(なし)	0 ーなし		○	○	
…タイトルに戻る	modoru		0	(なし)	0 ーなし		○	○	
…終了	syuuryou		0	(なし)	0 ーなし		○	○	

★毎月，読者の投稿プログラムのアイデアには目を見張るものがあります。(栃木県安蘇郡・**奥澤克典21歳**)……【**編**：ベーマガ採用基準の一番はアイデアですからね。あっと言わせる投稿待ってますよ。**影**：プログラム組めなくても，アイデアだけの投稿もOKです。】

するステージのタイムを３桁で書き込みます。そのすぐ後ろに“*”を入れてください。

その後は，パネルのデータ入力です。一つのパネルは３文字で表します。１文字目はX座標(横)，２文字目はY座標(縦)，３文字目はパネルの種類です。座標は１～15の範囲で表し，９より上は９，10，11，…ではなく，16進数表記で，９，a，b，c，…と表します。パネルの種類と対応する文字は，**第４表**を参照してください。この3文字による表記をパネルの数だけ繰り返し記述します。

それが終わったら，また“*”を入れます。最後は動かせるトラップパネルの数(トラップパネルの移動回数)を２桁で入力してください。

以上のデータをすべて１行で記述したものが，一つのステージとなっています。

プログラムの内容を理解できる方は，矢印パネルやトラップパネルの種類を増やしてみたりするのも，面白いと思います(例：矢印が突っ込むとそのまま消えてしまう，ERASEパネルなど)。また，スコアなどもつけてみては？

苦労した点

トラップパネルの移動方法を考えるのに，時間がかかりました。マウスポインタを十字にするのも初めてでしたし，ドラッグ＆ドロップのように動かすのも，いろいろと試行錯誤を繰り返しました。

でも，それ以外は意外とスムーズに進みました。

CHECKER FLAG

編：最初に画面を見たときは，2002年12月号掲載プログラムのHSP用『やじるしパズル ALAS!』のVB移植版かと思ってしまいました。

つぐ美：パズルゲームとしては，なかなかの完成度を誇る作品よ。

Dr.D：チョット気になった点としては，ゲーム画面の大きさじゃな。今のPCではあまり使用しない画面サイズなのかもしれんが，800×600サイズだと右側のコメントが途切れてしまうのじゃ。大きすぎる画面も考え物じゃな。

《リスト１》フォームのプログラム・リスト

```
Dim s$(11)
Dim px, py, c, xd, yd, p, k, st, tm, tit, stm, ch, ct, tr
Dim mm, md, mxy, mx, my, lx, ly, t1, t2, ea, eb, ec, ed, ow
Dim e$(3), b(500, 15, 15), kk(500), pp(500), MS$(3), m(17, 17)
Dim mt$, ag$, aa$(3)

'★★★　初期設定　★★★

Private Sub Form_Load()

With Form1    'フォームのプロパティ
    .Width = 14540
    .Height = 8870
    .AutoRedraw = True
    .Caption = "Panel Room"
    .Left = (Screen.Width - Form1.Width) ¥ 2  'フォームを画面の中央に表示
    .Top = (Screen.Height - Form1.Height) ¥ 2
    .BackColor = &H80000001
End With
With Picture1    'ピクチャのプロパティ
   .Width = 128
   .Height = 128
   .AutoRedraw = True
   .Visible = False
   .Picture = LoadPicture("panel.bmp")
   .BorderStyle = 0
End With
With Label1    'ラベルのプロパティ
   .Caption = "START"
   .FontSize = 30
   .Width = 122
   .Height = 35
   .Top = 429
   .Left = 420
   .BackColor = QBColor(9)
End With

For i = 0 To 16    '見えない「壁」つくる
  m(0, i) = 20: m(16, i) = 20: m(i, 0) = 20: m(i, 16) = 20
Next

'総ステージ数。追加するときはstmの値を増やし、コード冒頭のs$( )の値も同じにする
stm = 11

Open "data.txt" For Input As #1    'ステージデータ読み込み
  For i = 1 To stm: Input #1, s$(i): Next
  For i = 1 To 3: Input #1, MS$(i): Next
Close #1

st = 1: title    'タイトル画面へ
End Sub

'★★★　タイトル画面　★★★

Sub title()
Form1.Cls: tit = 1: ow = 0
Form1.MousePointer = 1
Form1.AutoRedraw = True
yarinaosi.Enabled = False
modoru.Enabled = False
modosu.Enabled = False

For i = 1 To 15: For J = 1 To 28    'マスの表示
  px = J: py = i: c = 0: hyouji
Next J, i

For i = -1 To 1: For J = -1 To 1    'タイトル文字
  Call moji(190 + J, 100 + i, "Ｐａｎｅｌ　Ｒｏｏｍ", 13, 70)
Next J, i
Call moji(190, 100, "Ｐａｎｅｌ　Ｒｏｏｍ", 5, 70)
Call moji(360, 270, "STAGE　" + Str$(st), 10, 40)
```

★ウチの県で毎月8日は八百屋の日らしい。／さいたま市は埼玉県と書いてある。なぜ？／「背表紙＝〇色」これで秘蔵グッズはもらったぜ！あれ？編さん？（長野市・**サバ柄14歳**）……【**編：**1月8日は“勝負事の日”らしい。イチかバチかというわけですね。】

```
Label1.Enabled = True: Label1.Visible = True    'ＳＴＡＲＴの表示
px = 14: py = 11: c = 6: hyouji: px = 15: py = 11: c = 8: hyouji '矢印の表示

Line (32, 32)-(928, 512), 0, B: Line (448, 352)-(512, 384), 0, B
Line (417, 417)-(543, 479), QBColor(9), BF
End Sub

'★★★  ゲーム画面の作成  ★★★

Sub gamen()
Line (32, 32)-(32 * 16, 32 * 16), 0, B: Line (576, 279)-(880, 281), 0, B
For i = -1 To 1: For J = -1 To 1
  Call moji(620 + J, 50 + i, "PanelRoom", 13, 35)
Next J, i
Call moji(620, 50, "PanelRoom", 5, 35)

px = 17: py = 10: c = 4: hyouji: px = 18: c = 9: hyouji: px = 19: c = 15: hyouji
Line (544, 320)-(640, 352), 0, B
Call moji(560, 358, "クリックするとその方向に進みます", 1, 15)
px = 17: py = 12: c = 1: hyouji: px = 18: c = 2: hyouji
Line (544, 384)-(608, 416), 0, B
Call moji(560, 422, "通過すると矢印の向きが90°変わります", 1, 15)
px = 17: py = 14: c = 3: hyouji
Line (544, 448)-(576, 480), 0, B
Call moji(560, 486, "通過すると矢印の向きが180°変わります", 1, 15)

stage
End Sub

'★★★  ステージの作成  ★★★

Sub stage()
For i = 1 To 15: For J = 1 To 15    'ステージ内容を白紙に
  m(J, i) = 0: px = J: py = i: c = 0: hyouji
Next J, i

tm = Val(Left$(s$(st), 3))    'タイムの読み込み
J = 0: p = 0
While Mid$(s$(st), J * 3 + 5, 1) <> "*"
  For i = 1 To 3    'パネルの座標、種類を読み込む
    e$(i) = Mid$(s$(st), i + J * 3 + 4, 1)
  Next i

  px = Val("&h" + e$(1)): py = Val("&h" + e$(2))  '読み込んだものを表示
  c = Val("&h" + e$(3)): hyouji: m(px, py) = c
  If c > 3 Then p = p + 1
  J = J + 1
Wend
k = Val(Right$(s$(st), 2))    '動かせるトラップパネルの数を読み込み
jotai
z = sndPlaySound("start", 1)
ct = 0: kiroku

Timer1.Enabled = True: Timer1.Interval = 1000  '制限時間カウントスタート
modosu.Enabled = False: yarinaosi.Enabled = False
End Sub
```

```
'★★★  キャラクターの表示  ★★★

Sub hyouji()
Form1.Print ""
BitBlt Form1.hDC, px * 32, py * 32, 32, 32, Picture1.hDC, (c Mod 4) * 32, Int(c / 4) * 32,
&HCC0020
End Sub

'★★★  マウスボタンを押す  ★★★

Private Sub Form_MouseDown(Button As Integer, Shift As Integer, x As Single, y As Single)
If tit = 0 Then
    mx = Int(x / 32): my = Int(y / 32)
    w = mx > 0 And mx < 16 And my > 0 And my < 16
    If w Then mxy = m(mx, my): mm = mxy Mod 4
    If mxy > 3 And w Then yajirusi
    If mxy > 0 And mxy < 4 And w Then trap
End If
End Sub

'★★★  矢印を押した時の処理  ★★★

Sub yajirusi()
xd = (mm - 1) * (mm < 3): yd = (2 - mm) * (mm > 0)  '矢印の方向
md = m(mx + xd, my + yd)                            '矢印が進む先の状態

If md > 3 Then
    If Int(md / 4) = Int(m(mx, my) / 4) Then        '矢印が同じ色だったら消す
        m(mx, my) = 0: c = 0: px = mx: py = my: hyouji
        m(mx + xd, my + yd) = 0: px = mx + xd: py = my + yd: hyouji
        p = p - 2
        z = sndPlaySound("erase", 1)
        kiroku
    End If
End If

If p = 0 Then clear    'パネルが全て消えたらクリア

If xd + yd <> 0 And md < 4 Then
    m(mx, my) = 0: px = mx: py = my: c = 0: hyouji    'トラップパネルを通過
    px = mx + xd: py = my + yd: ch = 0

    Select Case m(px, py)
      Case 1  '★  ＲＩＧＨＴ  ★
        mxy = mxy - 1: ch = 1: If mxy = 3 Or mxy = 7 Or mxy = 11 Then mxy = mxy + 4
      Case 2  '★  ＬＥＦＴ  ★
        mxy = mxy + 1: ch = 1: If mxy = 8 Or mxy = 12 Or mxy = 16 Then mxy = mxy - 4
      Case 3  '★  ＢＡＣＫ  ★
        mxy = mxy + 2: ch = 1: If mxy = 8 Or mxy = 9 Or mxy = 12 Or mxy = 13 Or _
                                  mxy = 16 Or mxy = 17 Then mxy = mxy - 4
    End Select
    If ch = 1 Then z = sndPlaySound("direction", 1)
    m(px, py) = mxy: c = mxy: hyouji: kiroku  '方向が変化したパネルを表示
End If
End Sub

'★★★  トラップパネルを押すとマウスポインタを十字に  ★★★
```

★音楽って素敵ですよね。(福井県武生市・つかち〜。**18歳**)……**【つぐ美：**影さんってステキ！**影：**ど，どうしたの。何をたくらんでるんだ？**つぐ美：**ただ，デート……。**影：**えーっ！デート？**つぐ美：**そう，デートがあるから今日の仕事お願いね，ということよ。**】**

```
Sub trap()
If k > 0 Then Form1.MousePointer = 5: tr = 1
End Sub

'★★★　ゲームの過程を記録（「一手戻す」の為に）★★★

Sub kiroku()
ct = ct + 1
For i = 1 To 15: For J = 1 To 15: b(ct, J, i) = m(J, i): Next J, i
kk(ct) = k: pp(ct) = p
modosu.Enabled = True: yarinaosi.Enabled = True
End Sub

'★★★　クリア　★★★

Sub clear()
Timer1.Enabled = False
z = sndPlaySound("clear", 1)
wait 2000
MsgBox "CLEAR !!", , "Panel Room"
st = st + 1: If st > stm Then ending    '次のステージへ
stage
End Sub

'★★★　タイトルの"START"上にポインタが無い場合文字の色は黒　★★★

Private Sub Form_MouseMove(Button As Integer, Shift As Integer, x As Single, y As Single)
Label1.ForeColor = 0
End Sub

'★★★　マウスボタンを離す　★★★

Private Sub Form_MouseUp(Button As Integer, Shift As Integer, x As Single, y As Single)
lx = Int(x / 32): ly = Int(y / 32)

If tit = 0 And tr = 1 Then    'トラップパネルの移動
    Form1.MousePointer = 1: tr = 0
    xd = Sgn(lx - mx): yd = Sgn(ly - my)
    If lx < 16 And ly < 16 And mx < 16 And my < 16 Then
        If xd + yd <> 0 And m(mx + xd, my + yd) = 0 And Abs(lx - mx + ly - my) = 1 And k > 0
Then
            px = mx: py = my: c = 0: hyouji: m(mx, my) = 0
            px = lx: py = ly: c = mxy: hyouji: m(lx, ly) = mxy: k = k - 1: jotai
            z = sndPlaySound("move", 1)
            kiroku
        End If
    End If
End If

If tit = 1 Then    'ステージの選択
    If lx = 14 And ly = 11 Then st = st - 1: If st < 1 Then st = stm
    If lx = 15 And ly = 11 Then st = st + 1: If st > stm Then st = 1
    title
End If
End Sub

'★★★　"ＳＴＡＲＴ"をクリックし、ゲーム画面へ　★★★

Private Sub Label1_Click()
Label1.Enabled = False
Label1.Visible = False
yarinaosi.Enabled = True
modoru.Enabled = True
Form1.Cls: tit = 0: gamen
End Sub

'★★★　"ＳＴＡＲＴ"上にポインタが重なると文字は黄色に　★★★

Private Sub Label1_MouseMove(Button As Integer, Shift As Integer, x As Single, y As Single)
Label1.ForeColor = QBColor(14)
End Sub

'★★★　タイトル画面へ戻る　★★★

Private Sub modoru_Click()
Form1.Cls: Timer1.Enabled = False: title
End Sub

'★★★　一手戻す　★★★

Private Sub modosu_Click()
ct = ct - 1: If ct < 1 Then ct = 1: Exit Sub
If ct = 1 Then modosu.Enabled = False
For i = 1 To 15: For J = 1 To 15    '一手戻した状態を表示
  m(J, i) = b(ct, J, i): px = J: py = i: c = m(J, i): hyouji
Next J, i
k = kk(ct): p = pp(ct): jotai
End Sub

'★★★　終了　★★★

Private Sub syuuryou_Click()
End
End Sub

'★★★　制限時間の経過　★★★

Private Sub Timer1_Timer()
tm = tm - 1: jotai: If tm = 0 Then timeover
End Sub

'★★★　ゲームのやり直し　★★★

Private Sub yarinaosi_Click()
stage
End Sub

'★★★　タイムオーバー　★★★

Sub timeover()
Timer1.Enabled = False
z = sndPlaySound("timeover", 1)
wait 2000
MsgBox "タイムオーバー", , "Panel Room"
```

★影さんこんにちは。12月号見ました！（神奈川県小田原市・**ハセ13歳**）……【**影：**あのね，"見ました"じゃないでしょ。毎月買ってね！ってお願いしたでしょ。**編：**見たってことは，買ったんじゃないでしょうか。**影：**それもそうだな。ま，今年もよろしく！】

```
title    'タイトル画面へ
End Sub

'★★★  文字の表示  ★★★

Sub moji(m1, m2, m3, m4, m5)
With Form1
    .CurrentX = m1
    .CurrentY = m2
    .FontSize = m5
    .ForeColor = QBColor(m4)
    .FontBold = True
End With
Form1.Print (m3)
End Sub

'★★★  ステージ数やタイムなどの情報を表示  ★★★

Sub jotai()
Form1.ForeColor = &H80000001
Line (770, 130)-(820, 155), , BF
Line (750, 180)-(800, 205), , BF
Line (860, 230)-(910, 255), , BF
Call moji(685, 130, "STAGE : " + Str$(st), 14, 15)
Call moji(685, 180, "TIME : " + Str$(tm), 14, 15)
Call moji(590, 230, "動かせるトラップパネルの数 : " + Str$(k), 14, 15)
End Sub

'★★★  ｗａｉｔ関数  ★★★

Function wait(ByVal w As Long)
t1 = GetTickCount: t2 = t1
While ((t2 - t1) < w): t2 = GetTickCount: DoEvents: Wend
End Function

'★★★  おわり  ★★★

Sub ending()
modosu.Enabled = False: yarinaosi.Enabled = False
st = 1: ea = 4: eb = 10: ec = 13: ed = 11: ow = 1
ag$ = "123456789abcdefghijk": mt$ = "3acefk4b6gj91725ih8d"
For i = 1 To 15: For J = 1 To 28: px = J: py = i: c = 0: hj2: Next J, i
Line (32, 32)-(928, 512), 0, B

For i = 1 To 20: For J = 1 To 3
  aa$(J) = aa$(J) + Mid$(MS$(J), InStr(mt$, Mid$(ag$, i, 1)), 1)
Next J, i

Call moji(120, 100, aa$(1), 14, 30)        '偉人からのメッセージ
Call moji(120, 180, aa$(2), 14, 30)
Call moji(270, 260, aa$(3), 15, 30)
Call moji(220, 420, "おめでとう！    ｂｙ  Ｈ. Ｉｎｏｕｅ", 13, 30)

Do
  For i = 1 To 28
    px = i: py = 1: c = ea: hj2: px = 29 - i: py = 15: c = eb: hj2
    wait 100: c = 0: hj2: px = i: py = 1: hj2
  Next i
  ea = ea + 4: If ea > 12 Then ea = 4
  eb = eb + 4: If eb > 14 Then eb = 6

  For i = 1 To 15
    px = 1: py = 16 - i: c = ec: hj2: px = 28: py = i: c = ed: hj2
    wait 100: c = 0: hj2: px = 1: py = 16 - i: hj2
  Next i
  ec = ec + 4: If ec > 13 Then ec = 5
  ed = ed + 4: If ed > 15 Then ed = 7

  DoEvents
Loop
End Sub

Sub hj2()
If ow = 1 Then
  Form1.AutoRedraw = False
  BitBlt Form1.hDC, px * 32, py * 32, 32, 32, Picture1.hDC, (c Mod 4) * 32, Int(c / 4) * 32,
&HCC0020
End If
End Sub
```

《リスト２》モジュールのプログラム・リスト

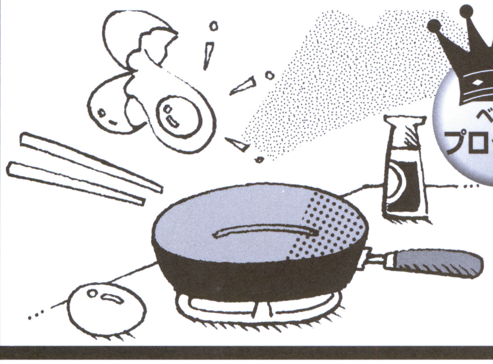
```
Option Explicit

Declare Function BitBlt Lib "gdi32" ( _
ByVal hDestDC As Long, _
ByVal x As Long, ByVal y As Long, _
ByVal nWidth As Long, ByVal nHeight As Long, _
ByVal hSrcDC As Long, _
ByVal xSrc As Long, ByVal ySrc As Long, _
ByVal dwRop As Long) As Long

Declare Function GetTickCount Lib "kernel32" () As Long

Declare Function sndPlaySound Lib "winmm.dll" Alias "sndPlaySoundA" _
(ByVal lpszSoundName As String, ByVal uFlags As Long) As Long
```

《リスト３》data.txtのリスト

```
050*664a676a5aa6*00
100*554881bb6*03
150*572774996b93*02
300*668a669718837916a7aa9*06
300*76767277997e88279c99aa959a1*05
350*76c678871975782981795893a9e9a8*05
350*661768967a6d678771971a7e69d791991a9a6a57ac9aaaa3*03
400*66a868a66772971684882a8d79199a6ad8abaa1*06
400*437538348644358 65e46256c87478188998f89eba4ca1ab2db3acddc5bd6cdc*07
500*479571677a75b7bc7f483589681a82b8fc8149e59469ca93b94c99*14
600*747943557855b5166fa6e47277297bc7e58c88bb88498791991c9a6a4aae5bf8b2bb77c49ca*11
に生、日に 学き、希。に昨今日びを望日明
な気ち忘れ。の持「なとの大？事はこい」を
アイシタイ ルンルン アート ベー ・ュ
```

★DDRメモリ512MB高いよねぇ。500円くらいで，ないかな〜。（大阪府枚方市・**松江祐輔21歳**）……【**編**：え？希望値の桁まちがえてない？**つぐ美**：512Mはバルクでも14,000円はするもんねぇ（注：2002年12月中旬現在）。**影**：500円じゃ，ベーマガも買えない！】

C++Builder V4.0用



卵を受け取って鍋に入れる

Flied Egg（フライド エッグ）

赤羊旅団（17歳）

ベストプログラム

ご挨拶

こんにちは，赤羊旅団です。

マウスとかトラックボールとか，人間とパソコンを結ぶものっていうのは，おもしろいですね。ゆくゆくは脳に直接つないで，体の一部分としてパソコンを扱う日が来るのでしょうが(？)，それまでは，マウスなどのデバイスの工夫は大切な問題ですよね。

そんなわけで，今回はマウスを利用したゲームを作ってみました。

内容

簡単に言うと，飛んできた卵をマウスでキャッチし，鍋に入れるゲームです。

ただし，モノが卵ですから，乱暴に扱うと割れてしまいます。マウスの先についているバーで，割れないようにやさしく受け止めて，的確に料理してください。

操作方法

マウスで画面上のバーを操作して，画面下部を横断している鍋の中に20個の卵を落とし込んでください。バーの操作方法は，以下のとおりです。

左クリック：バーを左に傾ける。

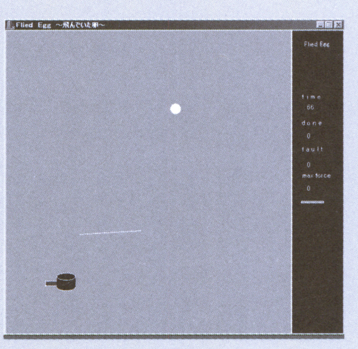
右クリック：バーを右に傾ける。

左右同時押し：バー周辺の卵を割る。

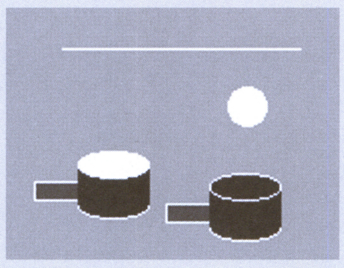
バーは，割れていない白い卵にのみ有効です。割れてしまった黄色い卵は，素通りします。

得点方法

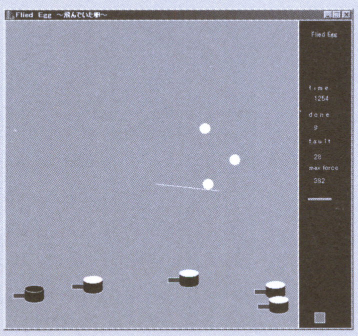
得点は，「入れた数，卵に与えた最大



《写真1》飛んでくる卵をうまく受け止め…



《写真2》鍋に落ちるように割って落としていく



《写真3》進行につれて卵・鍋の数も増加

《写真4》高得点を出すには，かなりの慣れが必要

★Dr.Dのプログラムページがびみょ～にリニューアル。掲載プログラムの改造が毎月紹介されるのは，楽しみです。（千葉県長生郡・Book On17歳）……【Dr.D：プログラムをいじってみることから，勉強は始まる。編：新「ベストプログラムへの道」ご意見お寄せください。】

の力の平均，時間」の要素を用いて，以下の計算で求めています。

総合点＝入れた数×100＋時間ボーナス ー平均力ーMISS×25(時間ボーナスは，制限時間内に20個クリアした時に与えられます)

仕組み

当たり判定は，バーの軌跡に当たる平行四辺形の辺々と卵の位置から計算しています。

どこかの辺をまたいでいると，当たったとみなしています。

摩擦のところは，バーと卵の垂直抗力を求め，それに摩擦係数FRICTIONをかけています。

改良

卵をやめてボールにして，バウンドするようにしてみたらおもしろいかもしれません。

もしくは，鍋の出現場所とマウスの移動範囲を変えて，バスケのフリーシュートみたいなゲームにしてもいいでしょうし。いろいろ改良できると思います。

その他

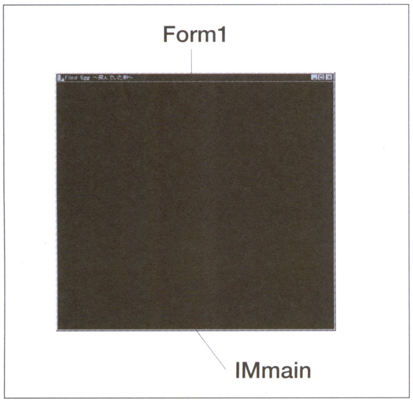
最高記録は2,800点ぐらい，友人の最低記録は−325,115です。むかついたらチャレンジしてみてください（笑)。

卵をキャッチするGameは数多くあれど，優しく捕まえる，というのはあまりないのではないでしょうか。

ベーマガに投稿したいと思えど，わが境遇それを許るさじ(といいつつ出してる受験生)。ツイニ，大学受験ナノデスネ。アア……。

たったこれだけのゲームに2ヶ月も費やしたのは，時間がなかったからです(たぶん)。辛いけど，また投稿するかもしれません。

お叱りありましたら yamada@nbn.ne.jp までよろしくお願いします。



《第1図》フォームの配置図

《第１表》プロパティ表

フォームの設定	
Name	Form1
AutoScroll	false
BorderIcons	[biSystemMenu,biMinimize]
BorderStyle	bsSingle
Caption	Flied Egg ～飛んでいた卵～
ClientHeight	580
ClientWidth	650
Color	clBlack
Height	607
Left	181
Top	122
Width	658
イベント	
OnClose	FormClose
イメージの設定	
Name	IMmain
Height	580
Left	0
Top	-8
Width	650
イベント	
OnMouseDown	IMmainMouseDown
OnMouseUp	IMmainMouseUp

CHECKER FLAG

Dr.D：卵の割れる判定や当たり判定など，プログラム面での工夫がしっかりなされておる作品じゃな。ただ，マウスの移動範囲も制限しておるため，ゲーム中に×ボタンクリックでの終了ができないゾ。一応Alt+F4キーで終了できるが，やはりESCキーによる終了処理などがほしいところじゃ。

影：改造の項目で卵をボールにしてみる，という改造を紹介しているけど，このゲームは卵が割れるという方が絶対に面白い！

編：ボールよりも卵を題材にしている方が，ユーモア感覚があって良いですよね。ボールでは一般的すぎます。

つぐ美：ちなみに，卵に向かってバーを上に移動させると，チョット面白いことになりますよ。

《リスト１》FliedEgg.cppのプログラム・リスト

```
//---------------------------------------------------------------------------
#include <vcl.h>
#pragma hdrstop
USERES("FliedEgg.res");
USEFORM("fliedEggUnit1.cpp", Form1);
//---------------------------------------------------------------------------
WINAPI WinMain(HINSTANCE, HINSTANCE, LPSTR, int)
{
        try
        {
                Application->Initialize();
                Application->CreateForm(__classid(TForm1), &Form1);
                Application->Run();
        }
        catch (Exception &exception)
        {
                Application->ShowException(&exception);
        }
        return 0;
}
//---------------------------------------------------------------------------
```

《リスト２》FliedEggUnit1.cppのプログラム・リスト

```
//---------------------------------------------------------------------------
#include <vcl.h>
#pragma hdrstop
#include <Math.h>
#include <stdlib.h>
double  GRAVITY=0.55,FRICTION=0.4,r=10,L=60,HARDNESS,mySin[90],myCos[90];
#define IMMAIN Form1->IMmain->Canvas
String label[]={" Flied Egg","","t i m e","d o n e","f a u l t","max force","作った数
","料理時間","失敗数","丁寧さ","総合得点！"};
#include "fliedEggUnit1.h"
#pragma package(smart_init)
#pragma resource "*.dfm"
TForm1 *Form1;TPLAYER *pl;
//---------------------------------------------------------------------------
// *-*-*-*-*  TForm1 の関数群  *-*-*-*-*
//              コンストラクタ
__fastcall TForm1::TForm1(TComponent* Owner) : TForm(Owner)
{  Application->OnIdle=AppIdleTitle; //まずはタイトル画面
   IMMAIN->Font->Color=clWhite;       //文字の色を白に
   pl=&player;titleFlag=true;         //pl→playerのポインタ
   randomize();
   for(int k=0;k<15;k++){vChara[k]=&egg[k];vChara[k+15]=&goal[k];}//アドレス格納
   for(int k=-45;k<45;k++){
     mySin[k+45]=sin(3.141592*k/180);//sinの値を得ておく
     myCos[k+45]=cos(3.141592*k/180);//cosの値を得ておく
}  }
//             右端のスコアを書く
void __fastcall TForm1::drawScore()
{
  IMMAIN->Brush->Color=clBlack;           //塗りつぶしの色＝くろ
  IMMAIN->Rectangle(550,0,650,580);       //画面塗りつぶし
  for(int k=0;k<6;k++){
    IMMAIN->TextOutA(570,20+k*50,label[k]);//各項目をかく
  }
  IMMAIN->TextOutA(580,140,time);         //スコアの表示
  IMMAIN->TextOutA(580,195,done);         //  〃
  IMMAIN->TextOutA(580,250,fault);        //  〃
  IMMAIN->TextOutA(580,295,force);        //  〃
  IMMAIN->Brush->Color=clRed;
  IMMAIN->Rectangle(580,570-F,600,570);   //力を赤い四角で描く
  IMMAIN->Rectangle(568,570-HARDNESS-2,612,570-HARDNESS+2);//割れる限界
}
//タイトル画面
void __fastcall TForm1::AppIdleTitle(TObject*Sender,bool &done)
{
  if(titleFlag==false){titleFlag=pl->down[0];return;}
  IMMAIN->Brush->Color=clBlack;
  IMMAIN->Rectangle(0,0,650,600);
  IMMAIN->Font->Size=35;
  IMMAIN->TextOutA(180,185,"ＦＬＩＥＤ　ＥＧＧ");
  IMMAIN->Font->Size=10;
  IMMAIN->TextOutA(240,270,"Please Click To Start !");
  if(pl->down[0]==true){
    pl->getVelocity(Left,Top+15); init();
    ClipCursor(&Rect(Left,Top+22,Left+Width-100,Top+500));
    Application->OnIdle=AppIdle;
} }
//      ＧＡＭＥ中のメイン関数
void __fastcall TForm1::AppIdle(TObject *Sender,bool &Done)
{
  IMMAIN->Brush->Color=(TColor)0x0090A4Ce; //色指定
  IMMAIN->Rectangle(0,0,550,580);          //画面消去
  if(done==20 || time==2600){              //クリア条件を満たせば
    drawRecord();ClipCursor(NULL);         //結果の表示。マウスの開放
    Application->OnIdle=AppIdleTitle;      //タイトルへ
  }
  if(time%700==0){lv++;HARDNESS-=8;}       //難易度を上げる
  if(F>maxForce){maxForce=F;}              //一番大きい力を記憶
  F*=0.75;time++;                          //力を減らす。時間を進める
  drawScore();                             //スコアの表示
  player.getVelocity(Left,Top+15);         //マウス位置から速度を求める
  player.draw(IMMAIN);                     //バー表示
  if(random(38)==0  && max[EGG_WAIT]<lv){makeChara(EGG_WAIT);}//卵発生
  if(random(32)==0  && max[GOAL_WAIT]<lv+2){makeChara(GOAL_WAIT);}//鍋発生
  for(int k=0;k<max[3];k++){           //キャラクタの総数回ループ
    chara[k]->move();                  //動かす
    chara[k]->draw();                  //描く
  }
  while(now+36>GetTickCount()){}       //処理が36ミリよりも早く終わると
  now=GetTickCount();                  //時間を取得
}
//キャラクタを登録する
void __fastcall TForm1::makeChara(ESTATE kind)
{
```

★最近のOFは，PC-98の話題で盛り上がっていますね。この調子で，ベーマガ本誌でもPC-98を取り上げてみては？（埼玉県さいたま市・や～まだ23歳）……【編：時代はもはやPC-98じゃないでしょ。創刊250号などの特集で扱うぐらいでしょうかね。Dr.D：うむ。】

```
  int k=kind*15+max[kind];   //種類に別の、さいごの配列をもとめる
  chara[max[3]]=vChara[k];   //空いているキャラノポイントを代入
  chara[max[3]]->init();     //初期化
  chara[max[3]]->reP=max[3]; //何番目に登録されたか記憶
  max[3]++;max[kind]++;      //総数を足す
}
//キャラクタを減らす
void __fastcall TForm1::deleteChara(TBASE* del,ESTATE kd)
{
  max[3]--;max[kd]--;del->state=kd;//数を減らし、準備状態にする
  vChara[max[kd]+kd*15]=del;       //空いているキャラクタとして代入
  chara[max[3]]->reP=del->reP;     //最後のキャラクタの番号を消したいキャラのものに
  chara[del->reP]=chara[max[3]];   //消したいキャラクタの場所に最後のキャラクタを代
入
}
//マウスの処理
void __fastcall TForm1::IMmainMouseDown(TObject *Sender,TMouseButton Button,
TShiftState Shift, int X, int Y)
{
 player.down[Button]=true;      //押されたボタンをＴＲＵＥに
 if(pl->down[0]+pl->down[1]==2){//もし両方押されていたら…
   if(time-pl->clickTime<2){    //同時ならば、卵破壊を
     int dist;                  //↓バーを表す直線
     line[0].set(mySin[pl->angle+45]/myCos[pl->angle+45],pl->mNowPt.x,pl->mNowPt.y);
     for(int k=0;k<15;k++){     //卵全体との当たり判定
       if(egg[k].state==EGG_FLY && fabs(pl->mNowPt.x-egg[k].X)<L){//位置を見て…
         dist=line[0].getDist(egg[k].X,egg[k].Y);//バーに近ければ…
         if(dist<20){egg[k].state=EGG_BROKEN;}   //壊す
 } } } }
 pl->clickTime=time;            //クリックした時間の更新
}
//マウスのボタンを離した時
void __fastcall TForm1::IMmainMouseUp(TObject *Sender, TMouseButton
Button,TShiftState Shift, int X, int Y)
{
 player.down[Button]=false;//離したボタンをＦＡＬＳＥに
}
//Ｇａｍｅをやめる時
void __fastcall TForm1::FormClose(TObject *Sender, TCloseAction &Action)
{
  ClipCursor(NULL);//一応マウスの開放を
}
//結果を書く
void __fastcall TForm1::drawRecord()
{
 int result=done*100+((time<2600)?1000:0)-fault*25-force*3;//総得点計算
 IMMAIN->Font->Size=34;                                    //文字の大きさ変更
 IMMAIN->TextOutA(70,100,"終了！ …… 結果発表");             //
 IMMAIN->Font->Size=13;                                    //文字の大きさ変更
 for(int k=6;k<11;k++){IMMAIN->TextOutA(140,k*30,label[k]);}   //各項目名書く
 IMMAIN->TextOutA(280,180,done*100);Sleep(500);IMmain->Refresh();//成功数
 IMMAIN->TextOutA(280,210,(time<2600)?1000:0);Sleep(500);IMmain->Refresh();//時間
 IMMAIN->TextOutA(280,240,-fault*25);Sleep(500);IMmain->Refresh();//失敗数
 IMMAIN->TextOutA(280,270,-force*3);Sleep(500);IMmain->Refresh();//丁寧さ
 IMMAIN->Rectangle(100,295,500,297);                       //ライン
 IMMAIN->TextOutA(280,300,result);IMmain->Refresh();Sleep(1500); //総合得点
 titleFlag=false;                                      //ゲーム状態をtitleに
}
//------------------------------------------------------------------------
//          *-*-*-*     TEGG の 関数群    *-*-*-*
//絵を書く
void __fastcall TEGG::draw()
{
  IMMAIN->Brush->Color=clYellow;                    //割れているなら黄色
  if(state==EGG_FLY){IMMAIN->Brush->Color=clWhite;} //割れてないなら白
  IMMAIN->Ellipse(X-r,Y-r,X+r,Y+r);                 //円を書く
}
//動かす
void __fastcall TEGG::move()
{ sY+=GRAVITY;              //重力を足す ↓バーの移動前移動後と、一致する直線を。
  line[0].set(mySin[pl->angle+45]/myCos[pl->angle+45],pl->mPrePt.x,pl->mPrePt.y);
  line[1].set(mySin[pl->angle+45]/myCos[pl->angle+45],pl->mNowPt.x,pl->mNowPt.y);
  if(state==EGG_FLY){      //卵が割れてなけりゃ。。。
    if( (line[0].getY(X)-Y)*(line[1].getY(X+sX)-Y-sY)<0 ||
line[1].getDist(X+sX,Y+sY)<r){
     //卵と直線の位置関係で、またいでいるかぶつかっていたら
       line[2].set((pl->sX!=0)?pl->sY/(pl->sX):100000,pl->mNowPt.x+myCos[pl-
>angle+45]*L,pl->mNowPt.y+mySin[pl->angle+45]*L);
       line[3].set((pl->sX!=0)?pl->sY/(pl->sX):100000,pl->mNowPt.x-myCos[pl-
>angle+45]*L,pl->mNowPt.y-mySin[pl->angle+45]*L);
      //↑バーの移動の軌跡の、側面の直線
      short wh=(sX>0)?2:3;            //卵の動いている方向
      int deb1=(line[wh].getY(X));    //動く方向と逆の側の直線との
      int deb2=(line[5-wh].getY(X+sX));//位置関係を調べる
      if( (deb1-Y)*(deb2-sY-Y)<=0 ){  //またいでいたら…
        double F,N,bs1,bs2,es1,es2;
```

★冬休みって何をしてますか？夏休みとちがって，休みもあっという間だし，寒いし，毎年，ごろごろしています。（京都市・**ボブサップイ15歳**）……【**Dr.D**：そういう時だからこそ，プログラムに明け暮れる。どうじゃ？**影**：手がかじかんで，キーが打てましぇん。】

```
        //バーに垂直な速度成分
        bs1=pl->sY*myCos[pl->angle+45]-pl->sX*mySin[pl->angle+45];
        //バーに平行な速度成分
        bs2=pl->sX*myCos[pl->angle+45]+pl->sY*mySin[pl->angle+45];
        es1=sY*myCos[pl->angle+45]-sX*mySin[pl->angle+45];//卵のバーに垂直な速度成分
        es2=sX*myCos[pl->angle+45]+sY*mySin[pl->angle+45];//卵のバーに平行な速度成分
        N=fabs(bs1-es1);F=fabs(bs2-es2);                //力を求める
        if(F>N*FRICTION){F=N*FRICTION;}     //摩擦を求める（最大摩擦力を超えない）
        es2+=((bs2-es2)!=0)?F*F/(bs2-es2):0;//速度差があれば、速度をあわせる
        double dist=line[0].getDist(X,Y);  //現在のバーの位置と卵の距離
        dist=((line[0].getY(X)-Y)>0)?dist-r:r-dist;//上か下か判断(getDistは絶対値)
        X+=dist*(-mySin[pl->angle+45]);            //位置をバーに合わせる
        Y+=dist*(+myCos[pl->angle+45]);            //位置をバーに合わせる
        sX=es2*myCos[pl->angle+45]-bs1*mySin[pl->angle+45];//速度をＸ方向になおす
        sY=bs1*myCos[pl->angle+45]+es2*mySin[pl->angle+45];//速度をＹ方向になおす
        F=F*F+N*N;                               //力計算(平方してないのはわざと)
        if(Form1->F<F){Form1->F=F;}                     //バーへの一番強い力を記憶
        if(F>HARDNESS){state=EGG_BROKEN;Form1->fault++;}//力が強いと割れる
      }}
  }X+=sX;Y+=sY;          //速度分動かす
  if(Y>580 || Y<-30){Form1->deleteChara(this,EGG_WAIT);Form1->fault++;}//画面外→消
す
}
//*-*-*-*-*  的  の  関数群    *-*-*-*-*
//卵とのあたり判定
void __fastcall TGOAL::checkEgg(TEGG *egg)
{
  for(int k=0;k<15;k++){
    if(fabs(egg[k].X-X)<30 && fabs(egg[k].Y-Y)<30){//Ｘ方向の位置を調べて…
      if(egg[k].state==EGG_BROKEN){                //卵が割れていれば
        Form1->deleteChara(&egg[k],EGG_WAIT);      //卵を消して
        Form1->setForce();                         //与えた力を計算して
        empty=false;Form1->done++;                 //卵を入れて、出来た数++
}}}}
//絵を書く
void __fastcall TGOAL::draw()
{
  IMMAIN->Brush->Color=clBlue;              //取っ手は青
  IMMAIN->Rectangle(X-40,Y+8,X-5,Y+18);     //取って描写
  IMMAIN->Brush->Color=clBlack;             //まわりは黒い色
  IMMAIN->Ellipse(X-18,Y+13,X+18,Y+27);
  IMMAIN->Pen->Color=clBlack;
  IMMAIN->Rectangle(X-18,Y,X+18,Y+20);
  IMMAIN->Pen->Color=clWhite;
  IMMAIN->Brush->Color=clYellow;            //卵あり＝中は黄色
  if(empty==true){
    IMMAIN->Brush->Color=clBlack;           //空=中は黒色
    checkEgg(egg);                          //卵との当たり判定
  }
  IMMAIN->Ellipse(X-18,Y-7,X+18,Y+7);       //上めんの描写
}
//動かす
void __fastcall TGOAL::move()
{
  X+=sX;Y+=sY;                                        //速度分動かす
  if(X<0 || X>540){Form1->deleteChara(this,GOAL_WAIT);}//画面外で消す
}
```

《リスト３》FliedEggUnit1.hのプログラム・リスト

```
//---------------------------------------------------------------------------
#ifndef fliedEggUnit1H
#define fliedEggUnit1H
#include <Classes.hpp>
#include <Controls.hpp>
#include <StdCtrls.hpp>
#include <Forms.hpp>
#include <ExtCtrls.hpp>
#include <Graphics.hpp>
//---------------------------------------------------------------------------
enum ESTATE{EGG_WAIT,GOAL_WAIT,EGG_FLY,EGG_BROKEN,MOVING};//状態の種類
class TLINE{  public:    //直線を表すクラス
 double A,B;             //Ｙ＝ＡＸ+Ｂ とした時の係数
 bool vertical;          //垂直かどうか(垂直はA,Bで表せない)
 double __fastcall getY(int x){  //Ｘに対応するＹを求める
   if(vertical==true){return (B>x)?-9999999:9999999;}
     return A*x+B;
 }
 double __fastcall getDist(int x,int y){//点と線の関係から
   return fabs(A*x-y+B)/sqrt(A*A+1);    //距離を出す
 }
 void __fastcall set(double k,double x,double y){//データ代入
   if(k>99999){vertical=true;B=x;}
   else{A=k;B=y-k*x;vertical=false;}
 }
};TLINE line[4];
```

★コンピュータ室の片隅に昔のベーマガ発見！1995年7月号と8月号，CD-ROMがついていたんですね。でも，中身はなかった。それからマイコンという雑誌も！いったい誰のでしょう。先生も知らないって言ってた。（岐阜県美濃市・**中村はじめ15歳**）……【**影：さあ？**】

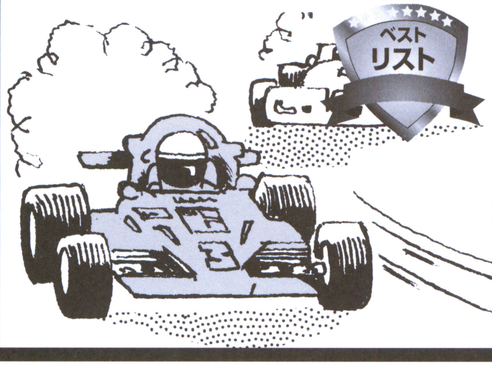
```
class TPLAYER{  public://プレイヤを
  POINT  mNowPt,//今のカーソル位置
         mPrePt;//前のカーソル位置（二つを用いて速度を計算）
  int    angle,//バーの角度
         clickTime;//クリックした時間（同時押しの判定）
  double sX,sY;//バーの動いた速度
  bool   down[2];//左右ボタンの状態（true = 押された）
  void __fastcall getVelocity(int fx,int fy){//速度を求める
    mPrePt=mNowPt;//今の位置を元の位置に
    GetCursorPos(&mNowPt);
    mNowPt.x-=fx;mNowPt.y-=fy;
    sX=mNowPt.x-mPrePt.x;
    sY=mNowPt.y-mPrePt.y;
  }
  void __fastcall draw(TCanvas* cnv){//絵を書く
    cnv->Pen->Color=clWhite;
    angle-=down[0]*3-down[1]*3;//ばーの回転
    if(fabs(angle)>44){angle+=down[0]*3-down[1]*3;}
    cnv->MoveTo(mNowPt.x-L*myCos[angle+45],mNowPt.y-L*mySin[angle+45]);
    cnv->LineTo(mNowPt.x+L*myCos[angle+45],mNowPt.y+L*mySin[angle+45]);
} };
class TBASE{      public://継承の基底
  double X,Y,//位置
         sX,sY;//速度
  int    time,reP;//計測用時間、chara[]の順番
  ESTATE state;//状態を記録
  virtual void __fastcall move(){};//動かす
  virtual void __fastcall draw(){};//描く
  virtual void __fastcall init(){}//初期化
};
class TEGG : public TBASE{   public://卵のクラス
  virtual void __fastcall move();
  virtual void __fastcall draw();
  virtual void __fastcall init(){
    X=random(2)*536+7;Y=random(150)+200;time=0;state=EGG_FLY;
    sX=(X<300)?random(6)+4:-random(6)-4;//←X速度 ↑位置とか
    sY=-random(12)-5;
  }
};TEGG     egg[15];
class TGOAL:public TBASE{    public://ゴールとなる鍋
 bool empty;//空いているかどうか
 void __fastcall checkEgg(TEGG*egg);//エッグとの当たり判定
  virtual void __fastcall move();
  virtual void __fastcall draw();
  virtual void __fastcall init(){   //初期化
    X=random(2)*520+15;Y=random(80)+470;sY=0;empty=true;
    sX=(X>300)?-2:2;state=MOVING;
} };TGOAL goal[15];
class TForm1 : public TForm
{
__published:    // IDE 管理のコンポーネント
        TImage *IMmain;
        void __fastcall IMmainMouseDown(TObject *Sender,
          TMouseButton Button, TShiftState Shift, int X, int Y);
        void __fastcall IMmainMouseUp(TObject *Sender, TMouseButton Button,
          TShiftState Shift, int X, int Y);
        void __fastcall FormClose(TObject *Sender, TCloseAction &Action);
private:    // ユーザー宣言
public:        // ユーザー宣言
        __fastcall TForm1(TComponent* Owner);
  void __fastcall AppIdle(TObject *,bool&);
  void __fastcall AppIdleTitle(TObject*,bool&);
  void __fastcall makeChara(ESTATE kind);//キャラクタを作る
  void __fastcall deleteChara(TBASE *,ESTATE);//キャラクタを消す
  void __fastcall drawScore();//今の経過成績
  void __fastcall drawRecord();//結果をかく
  void __fastcall setForce(){//卵に与えた力を更新する
    force=(force*done+maxForce)/(done+1);//平均を取る
    maxForce=0;
  }
  void __fastcall init(){//初期化
    done=fault=maxForce=F=force=time=0;lv=1;HARDNESS=250;
    for(int k=0;k<4;k++)max[k]=0;
    for(int k=0;k<15;k++){
      vChara[k]=&egg[k]; vChara[k+15]=&goal[k];
   } }
   TPLAYER  player;
   TBASE   *chara[30],*vChara[30];
   bool titleFlag;                //タイトル画面かどうか
   int now,time,max[4],lv,F;
   int force,maxForce,done,fault;
};//  速度調整用、プレイ時間用、キャラクタの数、プレイＬＶ、現在の力、
 //平均の力、最高の力、作った数、失敗した数
extern PACKAGE TForm1 *Form1;
//---------------------------------------------------------------------------
#endif
```

★復刊ドットコムが，復刊交渉に電波へ来た際には，ぜひその時の様子をベーマガで記事にしてほしい。（栃木県宇都宮市・**加藤従業員26歳**）……【**影**：その際はベーマガ広告交渉もさせてもらおう。**編集長**：ほお，影くんも仕事熱心じゃないか。**影**：おっ！ワンカップ！】

HSP V2.55用

レトロ感覚レース

レースdeぽん



寺内　洋平　(31歳)

内容・遊び方

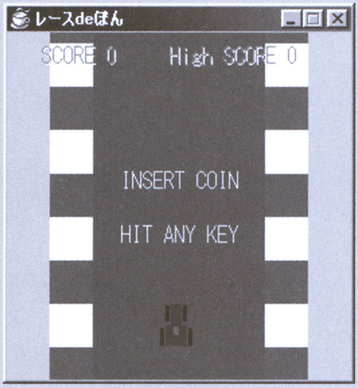
昔なつかし，カーレース『レースdeぽん』をHSPで制作しました。

HSPエディタでリストを入力し，F5キーを押して実行してください。

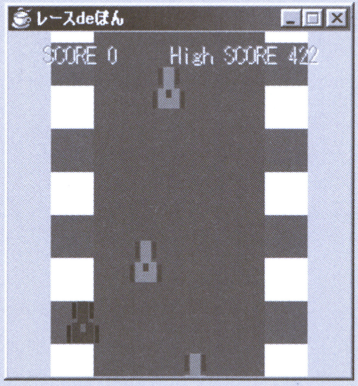
プログラム実行後，デモ画面になります(写真1)。「INSERT COIN」と表示されていますが，コインを入れる必要はありません。何かキーを押せばゲームスタートです。

遊び方は，ただ敵の車にぶつからないように進むだけです。1度でも敵の車にぶつかるとゲームオーバーです(写真4)。

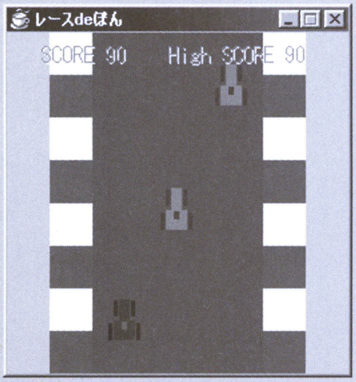
面クリアやゴールなどもありません。単純です。ですが，一応スコアだけはあります。進むごとにスコアが加算されていきます。ゲームオーバー後，ハイスコアが記録されます。ハイスコアは，ゲーム実行中のみ保存されています。



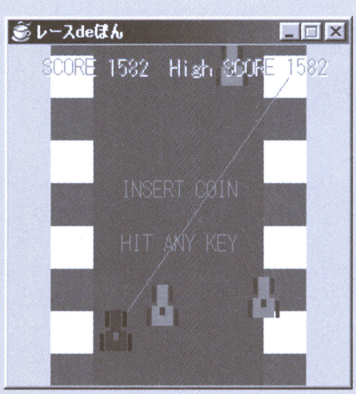
《写真1》タイトル画面でコインを挿入しないように！



《写真2》スタート時は，外側からコースインする



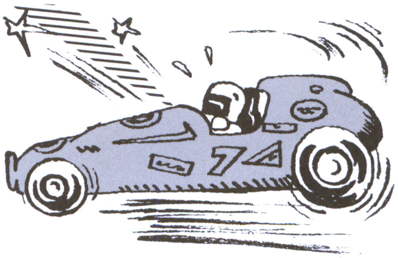
《写真3》敵車を抜くごとに得点が加算される



《写真4》接触してしまうと，その場でゲームオーバー

操作方法

スタート：何かキーを押す
左右移動：← →キー
スピードアップ：↑キー
スピードダウン：↓キー



CHECKER FLAG

影：ゲーム内容は，スタンダードというよりも，もはやレトロといった内容だね。

Dr.D：短いリストなのじゃが，もう少ししっかりしたコメント記述があれば，さらに良かったのぉ。

つぐ美：そのリスト容量やゲーム内容から，プログラム改造に向いている作品と言えますね。

編：ベーマガへ投稿の際には，もう少し説明文に記述があるとありがたいです。

★504i高いな〜と思っていたら，もう504iSだよ。さらに251iSでは3D液晶だって。進化早すぎ！（東京都立川市・自分は503iS19歳）
……【影：ベーマガは550円，なんて安いんだろう。中身は進化しても価格据え置き！読者のためのベーマガ，今年もよろしく！】

《リスト》レースdeぽんのプログラム・リスト

```
    title "レースdeぽん"
;
    gsel 0
    screen 0,256,256,1
    cls 4
    randomize
    redraw 0

    buffer 2,32,32,1
    cls 4
    color 0,0,255
    boxf  8, 1,10,12
    boxf 21, 1,23,12
    boxf  4,18, 6,31
    boxf 25,18,27,31
    color 255,0,0
    boxf 12,0,19,15
    boxf  8,16,23,31
    color 128,0,0
    boxf 14,17,17,22
;
    buffer 3,32,32,1
    cls 4
    color 32,0,0
    boxf  8, 1,10,12
    boxf 21, 1,23,12
    boxf  4,18, 6,31
    boxf 25,18,27,31
    color 0,0,255
    boxf 12,0,19,15
    boxf  8,16,23,31
    color 128,128,0
    boxf 14,17,17,22
;
    buffer 4,320,320,1
    cls 4
    color 0,255,0
    boxf   0,0, 31,319
    boxf 224,0,255,319
    l=0:COL=1
    *A
    if COL=0 :color 255,255,255:COL=1:goto *B
    if COL=1 :color 191,0,0    :COL=0
    *B
    boxf  32,l, 63,l+31
    boxf 192,l,223,l+31
    l=l+32
    if l<319 :goto *A
    color 0,180,0
    boxf 64,0,191,319
;
;    --- 変数の初期化 ---
    dim    X,20 : dim Y,20
    dim teki,20 : dim V,20
    X=110:RD=21:RL=32:HS=0
    l=1
    repeat 19
    X.i=0:Y.i=0:TEKI.i=0
    l=l+1
    loop
;
    screen 0,256,256,1
    pos 0,0
    cls 4
;
    *start
;
;    --- 変数の初期設定 ---
    SSP=3:SP=3:RD=400:BOMB=0:BOM=0:S=0:LV=0:BN=0
;
;    _*_*_*_ メインルーチン  _*_*_*_
;
    *main
    wait 1
;    --- 背景 ---
    pos 0,0:gmode 1:gcopy 4,0,RL,256,RL+256
    RL=RL-4 :if BOMB=0   :RL=RL-4
    if (SP=1) & (BOMB=0) :RL=RL+4
    if (SP=5) & (BOMB=0) :RL=RL-8
    if RL<1 :RL=64
;    --- 自分 ---
    C=0:D=0:if BOMB>0 :rnd C,7:C=C-3:rnd D,7:D=D-3:BOMB=BOMB+1
```

★ベーマガを買いつづけて5年目に突入！2003年，私も20歳になります。また一つDr.Dもお年をめされるわけだ。Dr.D，いつまでもお元気で。（鹿児島市・ビッグKAGE19歳）……【Dr.D：気遣いありがとう。じゃが，ワシはまだまだ若いもんには負けんぞ…ゴホゴホ。】

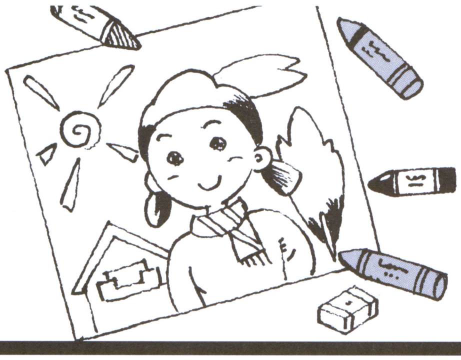
```
if BOMB=40 :BOMB=0:BN=0:palfade -60,0,255
if BOMB>0 :rnd RX,256:rnd RY,256:line X+16,216,RX,RY
pos (X+C),(200+D):gmode 2:gcopy 3,0,0,31,31
;    --- 敵車 ---
I=1:BOM=0
  repeat 19
   if TEKI.i=1           :gosub *kaku
   rnd R,RD
   if (R=1) & (TEKI.i=0) :gosub *deru
   if TEKI.i=1           :gosub *check
   I=I+1
  loop
;    --- ---- ---
if IC=0 :gosub *any
if IC=2 :color 255,255,255:pos 106,100:print "START!"
color 255,255,255
pos 26,9:print "SCORE "+S:pos 121,9:print "High SCORE "+HS
color 255,0,0
pos 25,8:print "SCORE "+S:pos 120,8:print "High SCORE "+HS
redraw 1:redraw 0
LV=LV+1:if LV=10 :LV=0:RD=RD-1:if RD<10 :RD=10
if BOM=1 :IC=0:BOMB=1
DR=DR+SP
stick KY,15
if (IC=0) & (KY!0) :if BOMB=0 :IC=2:palfade 0,0,255
if IC=2 :X=X-5:if X<1 :IC=1:goto *start
if IC=1 :gosub *car
goto *main

;    _*_*_ サブルーチン _*_*_

*deru
if DR<70 :return
TEKI.i=1
rnd X.i,96:X.i=X.i+64:Y.i=-31
DR=1
return

*kaku
if BOMB>0 :Y.i=Y.i-(SP*2):if Y.i<-18 :teki.i=0
Y.i=Y.i+SP:if Y.i>255 :TEKI.i=0:return
if I=BN :Y.i=Y.i+SP:X.i=X.i+VX
```

```
pos X.i,Y.i:gmode 2:gcopy 2,0,0,31,31
VR=(S/200)+3
rnd VI,VR
if VI<2 :rnd V.i,5:V.i=V.i-2
X.i=X.i+V.i
if X.i<64  :X.i=64  :V.i=-V.i
if X.i>160 :X.i=160 :V.i=-V.i
return

*any
rnd R1,256:rnd R2,256:rnd R3,256
color R1,R2,R3
pos 84,100:print "INSERT COIN":pos 84,140:print "HIT ANY KEY"
return

*car
if (KY&1) :if (X>61)  : X=X-1
if (KY&4) :if (X<162) : X=X+1
if (KY&1) :if (X>65)  : X=X-7
if (KY&4) :if (X<158) : X=X+7
SP=SSP
if (KY&2) :SP=5:;DR=DR+1
if KY=8 :SP=1
return

*check
if IC!1     :return
if Y.i<172  :return
if Y.i>230  :return
if X>59     :if SP=5 :S=S+1
if X>59     :if SP>1 :S=S+1:if HS<S :HS=S
if X.i+20<X :return
if X+20<X.i :return
BOM=1:BOMB=1:BN=I:SP=SSP
VX= 16
if X.i<X :VX=-16
return
;
;
;
stop
```

★最近のベーマガは誤植が多い気がします。1月号でも表紙・目次と実際掲載の74，75ページのプログラム言語の表記が違います。（群馬県桐生市・かの20歳）……【**編**：お詫びと訂正：両方本文のほうが正しいです。Visual C++ V6.0とHSP V2.5です。ごめんなさい。】

リストが誌面に掲載できない作品です。
BMホームページ(http://www.basicmagazine.net)からダウンロードしてお楽しみください。

HSP V2.55用



記録力と色彩感覚が試される

ぬりえ_ANIME（アニメ）

TOMOROW（22歳）

説明

アニメに自由に色を塗ればいいじゃん。見本のとおりに色を塗るゲーム。

操作方法

キャラが5人いるぞ。クリアすれば新しいキャラが選択できる。5人目は，どのキャラでもいいから満点とれば選べるおまけ。

キャラを選択したらゲームスタート。まずは，見本の色を覚える(**写真1**)。覚えたら左クリック。

ぬりえ画面が表示されたら色を塗る(**写真2**)。左上の白い紙に。マウスを使う。左クリックだ。右下の10個のボタンで色を選択。NEXT AND EXITというボタンで，色の判定を行う。規定の正解数以上ならクリア(**写真3**)。

左から4人クリアすればエンディング。5人目はおまけだから関係ない。

プログラム説明

パレット番号20～49に，1～30までの番号をふって，塗りわけられたパーツごとに色を変えてる。

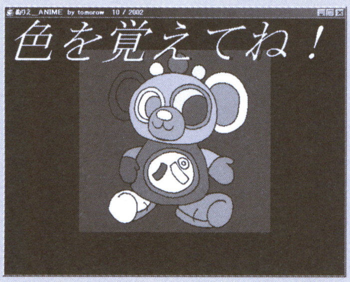
改造

パレットのさっき言った番号の，RGBのRの部分を1～30としてゲームで判定してるから，それを踏まえて絵だけ変えれば変わる。

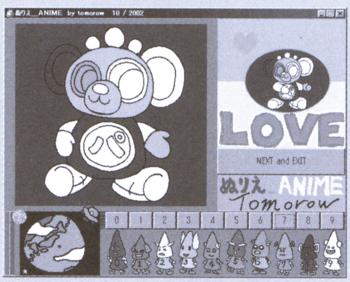
その他

JAVAで作ろうと思ったゲームだった。でも失敗。あきらめかけた。

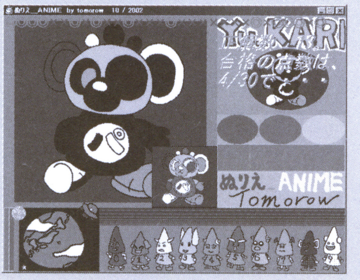
HSPというプログラム言語がタダで手



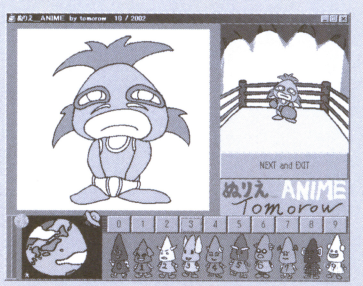
《写真1》まずは指定された配色を暗記



《写真2》記憶を頼りに色を塗っていく



《写真3》規定数以上の正解なら合格



《写真4》さまざまな塗り絵キャラクタを用意

★『ふわふわマルーン』楽しめます。Witchユーザーはプレイしてみて。1月号の掲載作品です。（神奈川県川崎市・.Local21歳）……【編：掲載プログラムに関する意見や感想が増えましたね。影：ワンダーウイッチ持ってること自慢したいんでしょう。くーっ！】

に入った，やった！簡単に作ろう，Windowsで初めてのプログラムだから。

HSPは1週間で覚えて，ゲームまで作ろうと思った。1か月かかった。やりたいことがやれなくて，実験を繰り返しできた。

グラフィックツールもタダのやつだ。全部タダで使えるものでゲームをつくってやった。

とりあえずゲームが作れた。これからもっとHSPを研究しなくちゃだめだと。

あと，マウスの右クリックは秘密だ。使うな！イライラってした時だけ，まあゆるす，使え！

★参考にしたもの

マイコンBASICマガジン 1997年12月号　FM-TOWNS用『ぬりえ』やみとも＆松瀬智宏　作

CHECKER FLAG

Dr.D：パレット機能をうまく使用して，色を塗る処理を実現させておる。ただ，そのパレット番号を指定している関係から，使用する画像(BMPファイル)の作成がチト面倒になってしまうな。

つぐ美：こういった色を区別するゲームの場合は，できるだけ似通った色がない方がありがたいわ。特にこのゲームは，色を覚えるのが趣旨であって，色を区別するゲームではありませんからね。

影：ゲームの進行具合を保存するセーブ機能がないのが残念。

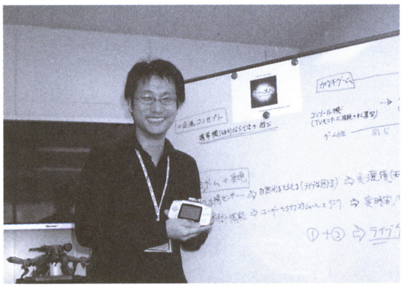
Pick up LIST

```
《リスト》塗り絵処理とマウス右クリック処理のリスト

*nurie  ;絵を塗り替える

x=mousex  : y=mousey ;座標入力
if x>=20  : else : goto *games
if y>=10  : else : goto *games
if x<=379 : else : goto *games
if y<=349 : else : goto *games

gsel sch  ;ウインドウID (sch:選択されたキャラ) へ
x=x-20 : y=y-10
pget x,y  ;点の色を取得

gsel 0    ;ウインドウID 0 へ
if rval=0 : goto *games ;線 (黒色0番) だったら何もしない
z=rval+19 ; 選択場所のパレット番号
palette z,r.iro,g.iro,b.iro : palfade : redraw ;現在の選択色で塗りつぶし

goto *games
;################
*irogae           ;マウスの右ボタンで選択色＋1
ckc=iro+1 : await 10 : mck=1 ; マウスチェック(mck)押しっぱなし
if ckc=10 : ckc=0
goto *games
stop
```

コナミが“太陽センサ”搭載の新機軸ゲームを発表

コナミは「メタルギアソリッド」などで有名な小島秀夫監督の手による新作アクションRPG『ボクらの太陽』（通称：ボクタイ）を発表しました。

同作品の特徴は，カートリッジに搭載されている「太陽センサ」と「時計機能」。「太陽センサ」とは，太陽から発する自然光に反応するセンサ（業界初）です。例えば，実際に，ゲーム機本体が太陽光を受けることで，ゲーム内の背景が明るくなったり，武器のエネルギーを補充できたり，ある特定の仕掛けが作動したりなど，実世界の光量や日付・時刻が，リアルタイムに反映されます。



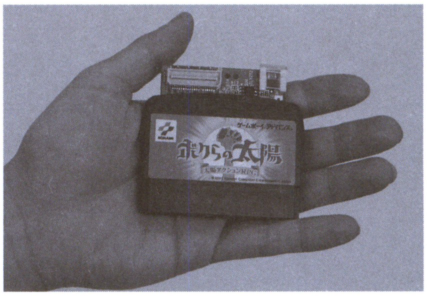
▲『ボクらの太陽』をデモンストレーションする小島秀夫監督

ゲームデザインを担当した小島監督によると，同作品のコンセプトは「天気や季節，といった自然界の要素を取り込んだ，ライブ感のあるゲーム」「携帯ゲーム機ならではの遊びを追求した，地球を感じるゲーム」とのことです。

日々変化する自然環境が攻略の鍵を握っているという，新機軸なこのゲーム。万が一，日光が出続けないための「太陽エネルギー貸し出し機能」（利用すると難易度が上がる），逆に，太陽光に浴び過ぎた場合の「プレイの中断を促す機能」が用意されているそうです。

「ボクらの太陽」は現在，2003年の発売に向けて開発中（監督自身，何度も屋外に出てテストプレイを繰り返しているそうです）。このソフトの登場で，新しいプレイスタイルが確立されるのか，今後が楽しみです。

（文／松原拓也）



▲「太陽センサ」を内蔵したゲーム『ボクらの太陽』（*写真は試作品です）

『ボクらの太陽』／コナミ
機種：ゲームボーイアドバンス専用
ジャンル：アクションRPG
発売予定時期：2003年春〜夏
価格未定

★250号もしくは256号までのカウントダウンは，やらないんですか（250号まであと○号ってやつ）？（広島県東広島市・ルミネ16歳）……【影：1・2・3・ダーッ！！……あれ？誰も乗ってくれないし…。ねね，あそぼぉー。一同：…（年末進行時のため忙しい)。】

降ってくる猫を捕獲
CatGet

キャット　　ゲット

宮崎　麦千代（17歳）

ストーリー

ある地域での出来事。とあるペットショップから猫が大量に逃げ出したらしい。その猫の捕獲を命令されたのが，新人の女の子バイト。それも1匹捕獲するごとに100円のバイト料と聞いては，やらないわけにはいかない。

ただし，逃げ出した猫と，そこらの野良猫が一緒に行動しているらしい。「捕獲するのは，逃げ出した猫だけだ。野良猫なんぞ捕獲したら1匹につきバイト料ー50円だ！」という店長の命令を受けて猫捕獲に向かうのだった。

内容

制限時間内に，上から降ってくる猫を捕獲していきます。大きい猫を捕獲すると，バイト料が＋100円，小さい猫を捕獲するとバイト料がー50円になります。

より多くのバイト料を稼いでください。最高金額を獲得すると，その金額が保存されます(**写真4**)。

操作方法

簡素なタイトル画面(**写真1**)でSTARTボタンを押すと，ゲームが開始します。

方向キーの左右で，キャラクタを移動させます。キャラクタが猫にぶつかると，自動的に猫を捕獲します。

Aキーを押すと，キャラクタがジャンプします(**写真3**)。小さい猫を避けたい時に使用します。

入力・実行方法

リスト入力は，メモ帳などのテキストエディタで行います。

リスト1を入力して“catget.c”という名前で保存します。同様に**リスト2**を入力して“img.c”という名前で保存します。

PIECEに付属しているコンパイラを使用して「pcc33 catget.c img.c」と入力してコンパイルを行います。

つぎにPCとPIECEを接続し，「run catget.srf」と入力すればPIECE上で実行できます。

なお，実行するとハイスコア保存用として，PIECE内に“cat_sc”というファイルが生成されます。

改造について

#defineで定義されている“TIME”の値を変更することで，残り時間を変更できます。

あと，同じように#defineで定義されている“JUMP_MAX”の値を変更することで，ジャンプの高さを変更できますが，あまり意味をなさないです。ちなみに，“JUMP_MAX”の値は偶数で指定してください。

《写真1》とにかく猫を捕まえるゲーム

《写真2》上から降ってくる猫を捕獲

《写真3》ジャンプのしすぎは，時間のロスになる

《写真4》はたしてバイト料は稼げるのか？

★12月号のOFに載った「背表紙の背景色」が，クイズになるとは思いませんでした。ただ，出題者である僕も，予想は外れてしまいましたが。（埼玉県さいたま市・**高橋博之15歳**）……【**編**：予想では“オレンジ”が一番人気でした。正解は“毎月変わる”でした。】

ついでに

キャラクタの移動に，微妙な慣性がついているのが，プログラム面での唯一の特徴です。使用しているキャラクタ画像は，「P/ECE HAND BOOK」のフリー素材集のものを使用させていただきました。

★参考にしたもの

・P/ECE HAND BOOK(まかべひろし著　JC2発行　電波新聞社発売)

CHECKER FLAG

編：単純な内容ですが，素材集の画像を使用しているので，見た目の印象が良いですね。

つぐ美：その素材集の中の「女の子_歩き」画像を利用して，女の子キャラクタの歩くアニメーションを表現しているわね。

Dr.D：慣性移動やジャンプは，物理学の公式などで動きを再現しているわけではないのじゃが，それなりの動きになっておるな。

《リスト1》catget.cのプログラム・リスト

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <piece.h>

//プロトタイプ
void game_init(int f);
void game_main(int f);
void cat_put(int f);
void cat_catch();
void chr_jump();
void put_gamedata();
void game_title();
void gameover();
void scoresave();

//画像関連
unsigned char vbuff[128*88];
extern unsigned char IMG[];
DRAW_OBJECT obj;
PIECE_BMP imgbmp;

//ゲーム関連の定数&変数
#define LEFT 0
#define RIGHT 1
#define JUMP_MAX 44
#define AC_MAX 35
#define TIME 1500
int chr_x, chr_y, jump;
int cat_x[2], cat_y[2], catch_f;
long score=0, hiscore=0, time=-1;

//PIECE本体関連の初期化
void pceAppInit( void )
{
  unsigned char *vram;

  memset( vbuff, 0, 128*88 );
  vram = pceLCDSetBuffer(INVALIDPTR);
  pceAppSetProcPeriod( 20 );

  pceLCDDispStop();
  pceLCDSetBuffer( vbuff );
  pceLCDDispStart();

  //ゲーム関連初期化
  game_init(0);
}

//ゲームメイン実行部分
void pceAppProc( int cnt )
{
  //タイトル
  if(time == -1)
    game_title();

  //ゲーム中
  if(time > 0)
    game_main(1);

  //ゲーム終了
  if(time == 0)
    gameover();

  //画面転送
  pceLCDTrans();
}

//終了時
void pceAppExit( void )
{}

//ゲーム関連の初期化
void game_init(int f)
{
  PBMP_FILEHEADER pbhead;
  PCETIME ptime;
  FILEACC pfa;
  int i,j;

  if(f==0){
   //画像読み込み
   memcpy(&pbhead, IMG, sizeof(PBMP_FILEHEADER));
    imgbmp.header = pbhead;
    imgbmp.buf = IMG+sizeof(PBMP_FILEHEADER);
```

```
    imgbmp.mask = IMG+sizeof(PBMP_FILEHEADER)+(imgbmp.header.h * imgbmp.header.w)/4;

    // ファイルオープン
    if( (pceFileOpen(&pfa,"cat_sc",FOMD_RD) == 1) ){
      //ファイル生成
      hiscore = 1000;
      pceFileCreate("cat_sc",sizeof(int)+1);
      pceFileOpen(&pfa,"cat_sc",FOMD_WR);
      pceFileWriteSct(&pfa,&hiscore,0,sizeof(long));
      pceFileClose(&pfa);
    }
    //ハイスコア読み込み
    pceFileOpen(&pfa,"cat_sc",FOMD_RD);
    pceFileReadSct(&pfa,&hiscore,0,sizeof(long));
    pceFileClose(&pfa);
  }

  //画面初期化
  memset( vbuff, 0, 128*88 );

  //地面を描画しておく
  for(i=0; i < 2; i++){
  for(j=0; j < 8; j++){
    pceLCDSetObject(&obj, &imgbmp, j*16,(i*8)+72, 0,16, 16,8, DRW_NOMAL);
    pceLCDDrawObject(obj);
  }
  }

  //randomize()処理
  pceTimeGet(&ptime);
  srand((unsigned int)ptime.ss);

  //変数初期化
  chr_x = 56; chr_y = 56; jump = 0; score = 0; catch_f = 1;
  cat_y[0]=0; cat_y[1]=0;
  game_main(0);
}

void game_main(int f)
{
  long lKey=pcePadGet();
  int chr_bmp_x[4] = {32,48,64,80};
  int chr_bmp_y[2] = {0,16};
  static int chr_f, chr_no, f_cnt, ac_r, ac_l;

  if(f==0){ //初期化
    chr_f = RIGHT; chr_no = 0; f_cnt = 0; ac_r = 0; ac_l = 0;
  }

  else{
    //画面初期化
    memset( vbuff, 0, 128*72 );
```

```
    //左右のキー入力
    if(lKey & PAD_RI){
      if(chr_f == LEFT){
        chr_f = RIGHT;
      }
      //慣性の動き
      if(ac_l > 0){
        ac_l-=2;
        if(ac_l < 0)
          ac_l = 0;
      }
      else if(ac_r < AC_MAX)
        ac_r++;
    }
    if(lKey & PAD_LF){
      if(chr_f == RIGHT){
        chr_f = LEFT;
      }
      //慣性の動き
      if(ac_r > 0){
        ac_r-=2;
        if(ac_r < 0)
          ac_r = 0;
      }
      else if(ac_l < AC_MAX)
        ac_l++;
    }

    //右移動
    if(ac_r > 0 ){
      //アニメーション用
      f_cnt++;
      if(f_cnt == 6){
        f_cnt=0;
        chr_no++;
        chr_no%=4;
      }
      if(chr_x < 112){
        chr_x++; //座標移動
      }
    }

    //左移動
    if( ac_l > 0){
      //アニメーション用
      f_cnt++;
      if(f_cnt == 6){
        f_cnt=0;
        chr_no++;
        chr_no%=4;
      }
```

★256号は本当に盛大にやるんですか？（東京都八王子市・**金さん34歳**）……【**編：**その前に250号があります。256号とあわせて楽しみにしてください。**影：**本当だな，楽しみにしてるぞ。**編：**影さんもがんばるんですよ。**影：**本当だな，私中心でいいんだな。】

```
    if(chr_x > 0){
      chr_x--; //座標移動
    }
  }

  //ジャンプ
  if( (IKey & TRG_A) && jump == 0){
      jump = JUMP_MAX;
  }
  //キャラクタジャンプ処理
  chr_jump();

  //キー入力がないとき
  if(IKey == 0){
    //慣性の動き
    if(ac_r > 0){
      ac_r--;
    }
    else if(ac_l > 0){
      ac_l--;
    }
    //何もしていない時のキャラクタ(ジャンプ時含む)
    else{
      if(chr_f == LEFT){
        if(jump < JUMP_MAX/2 && jump!=0)
          chr_no=0;
        else
          chr_no=1;
      }
      if(chr_f == RIGHT){
        if(jump < JUMP_MAX/2 && jump!=0)
          chr_no=1;
        else
          chr_no=0;
      }
      f_cnt=0;
    }
  }

  //猫の位置設定＆表示
  if(catch_f==1){
    cat_put(0);
  }
  cat_put(1);

  //猫を捕まえたか判定
  cat_catch();

  //キャラクタ表示
  pceLCDSetObject(&obj, &imgbmp, chr_x, chr_y,
        chr_bmp_x[chr_no], chr_bmp_y[chr_f],
        16,16, DRW_NOMAL);
```

```
    pceLCDDrawObject(obj);

    //タイム処理
    time--;

    //ゲーム関連のパラメータ表示
    put_gamedata();
  }
}

//猫を表示
void cat_put(int f)
{
  int f2, cnt;
  static int xf=0;

  //座標設定
  if(f == 0){
    //画面上から落下
    cat_y[0]+=2; cat_y[1]+=2;

    //猫１ X座標設定
    if(xf == 0){
      cnt = 0;
      do{
        f2 = 0; cat_x[0] = rand()%112;
        if(cat_x[0] >= chr_x-16 && cat_x[0] <= chr_x+16){
          f2 = 1; cnt++;
        }
        //万が一の無限ループ回避用
        if(cnt == 50){
          f2 = 1; cnt = 0; cat_x[0] = 0;
        }
      }while(f2==1);

      //猫２ X座標設定
      cnt = 0;
      do{
        f2 = 0; cat_x[1] = rand()%112;
        if(cat_x[1] >= chr_x-16 && cat_x[1] <= chr_x+16){
          f2 = 1; cnt++;
        }
        if(cat_x[1] >= cat_x[0]-16 && cat_x[1] <= cat_x[0]+16){
          f2 = 1; cnt++;
        }
        //万が一の無限ループ回避用
        if(cnt == 50){
          f2 = 1; cnt = 0; cat_x[1] = 116;
        }
      }while(f2==1);
      xf = 1;
    }
```

```
  }

  //地面到着(Y座標決定)
  if(cat_y[0] == 56){
    catch_f = 0; xf = 0;
  }

  //猫１表示
  pceLCDSetObject(&obj, &imgbmp, cat_x[0], cat_y[0], 0, 0, 16,16, DRW_NOMAL);
  pceLCDDrawObject(obj);

  //猫２表示
  pceLCDSetObject(&obj, &imgbmp, cat_x[1], cat_y[1], 16, 0, 16,16, DRW_NOMAL);
  pceLCDDrawObject(obj);
}

//猫キャッチ判定
void cat_catch()
{
  int i;

  for(i=0; i<2; i++){
    if( ((chr_x+6) >= cat_x[i]) && ( chr_x <= (cat_x[i]+6))){ //X座標チェック
      if(chr_y == cat_y[i]){ //Y座標チェック
        catch_f = 1; cat_y[0]=0; cat_y[1]=0;
        //得点配分
        if(i == 0)//猫１(大きい)
          score+=100;
        else //猫２(小さい)
          score-=50;
      }
    }
  }
}

//キャラクタジャンプ処理
void chr_jump()
{
  static int count = 10;

  if(jump > JUMP_MAX/2) //上へ
    chr_y--;
  else if(count > 0){ //一定時間, 空中で平行移動
    count--; jump++;
  }
  else if(jump > 0) //下へ
    chr_y++;

  if(jump > 0)
    jump--;
  if(jump == 0)
    count = 10;
}

//スコア＆タイム表示
void put_gamedata()
{
  pceFontSetType(0);
  pceFontSetPos( 0, 0);
  pceFontPrintf("    バイト料:%4d円",score);
  pceFontSetPos( 0, 10);
  pceFontPrintf("過去の最高額:%4d円",hiscore);
  pceFontSetPos( 0, 20);
  pceFontPrintf("残り時間:%d",time);
}

//タイトル表示
void game_title()
{
  long lKey=pcePadGet();

  pceFontSetType(1);
  pceFontSetPos( 36, 15);
  pceFontPrintf("Cat Get");

  pceFontSetPos( 8, 35);
  pceFontPrintf("Push START Key");

  if( lKey & PAD_C)
    time = TIME;
}

//ゲームオーバー時の処理
void gameover()
{
  long lKey=pcePadGet();
  static int f = 0;

  if(score >= hiscore){
    pceFontSetType(0);
    pceFontSetPos( 41, 50);
    pceFontPrintf("Hi-Score!!");
  }

  pceFontSetType(1);
  pceFontSetPos( 28, 30);
  pceFontPrintf("GAME OVER");

  if(f == 0){
    if(score >= hiscore){
      scoresave();
    }
  }
```

★こんにちはー！私は15歳ただいま彼氏募集中。（神奈川県小田原市・**サヤカ15歳**）……【**つぐ美：**女の子からの投稿です。うれしいわ。プログラムとかやってるのかな？今度そのへんの話もしてちょうだいね。待ってるわよ。**影：**また小田原市からの投稿か。うーん。】

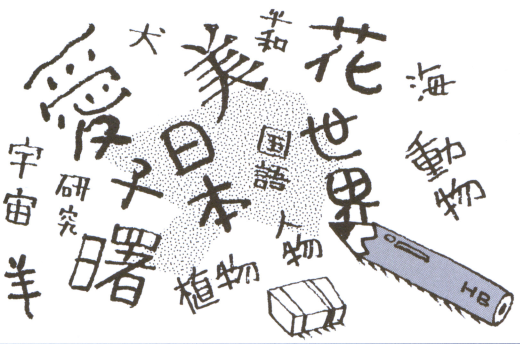
```
    if( (IKey & PAD_A) || (IKey & PAD_B)){
      game_init(1);
      time = -1; f = 0;
    }
  }

  //スコア保存
  void scoresave()
  {
    FILEACC pfa;

    hiscore = score;
    pceFileOpen(&pfa,"cat_sc",FOMD_WR);
    pceFileWriteSct(&pfa,&hiscore,0,sizeof(long));
    pceFileClose(&pfa);
  }
```

《リスト2》img.cのプログラム・リスト

```
//キャラクタの画像データ
const unsigned char  IMG[] = {
   80,  77,  66,  80, 148,   4,   0,   0,   2,   1,  96,   0,  32,   0,   0,   0,
  128,   4,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,
    0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,  63, 192,   3, 252,
    0,   0,   0,   0,   0,  15, 255, 192,   0,  15, 255,   0,   0,  15, 255, 192,
    0,  15, 255,   0,  53, 127, 253,  92,   0,   0,   0,   0,   0,  53, 205,  48,
    0,  53, 205, 192,   0,  53, 205,  48,   0,  53, 205, 192,  53,  89, 101,  92,
    0,   0,   0,   0,   0, 213, 117,  92,   0, 213, 116, 112,   0, 213, 117,  92,
    0, 213, 116, 112,  57,  89, 101, 108,   0,   0,   0,   0,   0, 221,  94, 151,
    0, 221,  93,  92,   0, 221,  94, 151,   0, 221,  93,  92,  13,  85,  85, 112,
    0,   0,   0,   0,   0, 243, 151, 252,   0, 243, 158,  92,   0, 243, 151, 252,
    0, 243, 158,  92,  14,  85,  85, 176,   3, 192,  60,   0,   0, 204, 245, 192,
    0, 204, 247, 156,   0, 204, 245, 192,   0, 204, 247, 156,  13,  85,  85, 112,
    3, 191, 236,   0,   0, 204,  54, 192,   0, 204,  54, 240,   0, 204,  54, 192,
    0, 204,  54, 240,  61, 117,  93, 124,   3, 170, 172,   0,   0,  48, 251,   0,
    0,  48, 251,   0,   0,  48, 251,   0,   0,  48, 251,   0,  13,  85,  85, 112,
    2, 186, 232,   0,   0,  15, 220,   0,   0,  15, 220,   0,   0,  15, 220,   0,
    0,  15, 220,   0,  51,  85,  85, 204,   3, 170, 172,   0,   0,  15,  59,   0,
    0,  14, 204,   0,   0,  14, 179,   0,   0,  14, 204,   0,   0, 255, 255,   0,
    0, 234, 176,   0,   0,  60, 235,   0,   0,  59,  59,   0,   0,  58, 172, 192,
    0,  59,  59,   0,   0,  53,  85, 192,   0,  63, 192, 192,   0,  59, 175, 192,
    0,  58, 235,   0,   0,  62, 171, 192,   0,  58, 235,   0,   0,  53,  85, 115,
    0,  58, 179,   0,   0,  63, 247,   0,   0,  15, 252,   0,   0,  55, 255, 192,
    0,  15, 252,   0,   0,  54, 214, 124,   0,  58, 172,   0,   0,  13, 240, 192,
    0,   3,  12,   0,   0, 193, 247,   0,   0,   3,  12,   0,   0,  53, 215, 112,
    0,  59, 188,   0,   0,  59,  58, 192,   0,  14, 172,   0,   0, 235,  10, 192,
    0,  14, 172,   0,  85,  85,  85,  86,   0,   0,   0,   0,   0,   0,   0,   0,
    0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,  85,  86,  89,  85,
    0,   0,   0,   0,   0,  15, 255,   0,   0,  63, 255,   0,   0,  15, 255,   0,
    0,  63, 255,   0,  89,  85,  85,  85,   0,   0,   0,   0,   0,  55,  53, 192,
    0, 199,  53, 192,   0,  55,  53, 192,   0, 199,  53, 192,  85,  85,  85,  85,
    0,   0,   0,   0,   0, 209, 213, 112,   3,  85, 213, 112,   0, 209, 213, 112,
    3,  85, 213, 112,  85,  85,  85,  89,   0,   0,   0,   0,   3,  87,  87, 112,
   13, 107,  87, 112,   3,  87,  87, 112,  13, 107,  87, 112,  85,  89, 101,  85,
    0,   0,   0,   0,   3,  91, 108, 240,   3, 253, 108, 240,   3,  91, 108, 240,
    3, 253, 108, 240, 101,  85,  85,  85,   0,   0,   0,   0,   3, 109, 243,  48,
    0,  53, 243,  48,   3, 109, 243,  48,   0,  53, 243,  48,  85,  85,  85,  85,
    0,   0,   0,   0,   0, 249, 195,  48,   0,  57, 195,  48,   0, 249, 195,  48,
    0,  57, 195,  48,  85,  85,  85,  85,   0,   0,   0,   0,   0,  14, 240, 192,
    0,  14, 240, 192,   0,  14, 240, 192,   0,  14, 240, 192,   0,   0,   0,   0,
    0,   0,   0,   0,   0,   3, 127,   0,   0,   3, 127,   0,   0,   3, 127,   0,
    0,   3, 127,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   3,  59,   0,
    0,  12, 235,   0,   0,   3,  59,   0,   0,  14, 207,   0,   0,   0,   0,   0,
    0,   0,   0,   0,   0,  14, 206, 192,   0,  51, 170, 192,   0,  14, 206, 192,
    0,  14, 179, 192,   0,   0,   0,   0,   0,   0,   0,   0,   0,  14, 186, 192,
    0,  62, 171, 192,   0,  14, 186, 192,   0,  63, 174, 192,   0,   0,   0,   0,
    0,   0,   0,   0,   0,   3, 255,   0,   0,  63, 253, 192,   0,   3, 255,   0,
    0,  13, 255, 192,   0,   0,   0,   0,   0,   0,   0,   0,   0,   3,  12,   0,
    0,  13, 244,  48,   0,   3,  12,   0,   0,  48, 247,   0,   0,   0,   0,   0,
    0,   0,   0,   0,   0,   3, 171,   0,   0,  58,  14, 176,   0,   3, 171,   0,
    0,  58, 206, 192,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,
  120,  30,   0,   0,   3, 248,   3, 240,   3, 248,   3, 240, 127, 254,   0,   0,
    7, 252,   7, 248,   7, 252,   7, 248, 127, 254,   0,   0,  15, 254,  15, 252,
   15, 254,  15, 252, 127, 254,   0,   0,  15, 255,  15, 254,  15, 255,  15, 254,
   63, 252,   0,   0,  15, 254,  15, 254,  15, 254,  15, 254,  63, 252,  24,  96,
   15, 248,  15, 254,  15, 248,  15, 254,  63, 252,  31, 224,  15, 248,  15, 252,
   15, 248,  15, 252, 127, 254,  31, 224,   7, 240,   7, 240,   7, 240,   7, 240,
   63, 252,  31, 224,   3, 224,   3, 224,   3, 224,   3, 224,  95, 250,  31, 224,
    3, 240,   3, 224,   3, 240,   3, 224,  15, 240,  15, 192,   7, 240,   7, 240,
    7, 248,   7, 240,   7, 248,   7, 136,   7, 248,   7, 240,   7, 248,   7, 240,
    7, 253,   7, 208,   7, 240,   3, 224,   7, 248,   3, 224,   7, 254,   7, 224,
    3, 248,   1, 224,  15, 240,   1, 224,   7, 252,   7, 224,   7, 120,   3, 224,
   15,  56,   3, 224, 255, 255,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,
  255, 255,   0,   0,   3, 240,   7, 240,   3, 240,   7, 240, 255, 255,   0,   0,
    7, 248,  15, 248,   7, 248,  15, 248, 255, 255,   0,   0,  15, 252,  31, 252,
   15, 252,  31, 252, 255, 255,   0,   0,  31, 252,  63, 252,  31, 252,  63, 252,
  255, 255,   0,   0,  31, 252,  31, 252,  31, 252,  31, 252, 255, 255,   0,   0,
   31, 252,   7, 252,  31, 252,   7, 252, 255, 255,   0,   0,  15, 252,   7, 252,
   15, 252,   7, 252, 255, 255,   0,   0,   3, 248,   3, 248,   3, 248,   3, 248,
    0,   0,   0,   0,   1, 240,   1, 240,   1, 240,   1, 240,   0,   0,   0,   0,
    1, 240,   3, 240,   1, 240,   3, 240,   0,   0,   0,   0,   3, 248,   7, 248,
    3, 248,   3, 248,   0,   0,   0,   0,   3, 248,   7, 248,   3, 248,   7, 248,
    0,   0,   0,   0,   1, 240,   7, 248,   1, 240,   3, 248,   0,   0,   0,   0,
    1, 224,   3, 252,   1, 224,   7, 240,   0,   0,   0,   0,   1, 240,   7,  60,
    1, 240,   7, 184
};
```

★ドライバ1本使わない影先生の新奥義，恐れ入りました。次はぜひウチのPCをギャンにしてください。あ，ズゴックでも。（愛媛県新居浜市・**飛べ，電波ビル製ホワイトベース！：赤羊旅団18歳**）……【**影：**ぐはははは，2002年12月号のあの技だな。よしよし。】

iアプリ用



四字熟語を知らないと辛いゾ

四字熟語探し（よんじじゅくごさが）

佃　貴弘（24歳）

ゲーム内容

8×6のマスから計12個の四字熟語を見つけだすゲームです。

操作方法

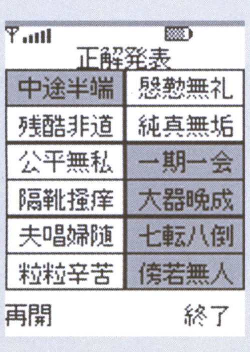
方向キーと選択キーを使って，マスの中から漢字を四つ選択し，四字熟語を作成していきます。もし，間違えたときは，ソフトキー1を押すことで選択をやり直すことができます。

漢字を順番に四つ選択して熟語を作成すると，その熟語が正しいかどうかが判定されます(**写真1**)。

熟語の間違いをコンピュータに3回指摘されるかソフトキー2を押すと，正解が発表されます(**写真3**)。このとき選択キーかソフトキー1を押すと新たにゲームを始め，ソフトキー2を押すとゲームを終了します。

《写真1》漢字を組み合わせて四字熟語を作る

《写真2》数が少ない方がかえって難しい？



《写真3》四字熟語に熟知していないとクリアは難しい

《写真4》実機がない人は，エミュレータで楽しもう

プログラム構造について

今回のプログラムは，キー入力部分とメインプログラム部分に分かれます。この二つが独立して実行され，キーが押されたときにix,iy,se,sk1,sk2の数値を変更するように割り込み処理をします。キー入力以外の処理は，paintメソッドの中にあります。

プログラム制作に当たって

今回は，gif画像をまったく使わずにゲームを作成することにこだわってみました。gifを使うと，直接打ち込んでプログラムの勉強をする人には不向きだと考えたからです。

けれども，iアプリの場合，使用機種によって画面やフォントのサイズが異なっているので，その修正のために，些細な計算をしなければなりませんでした。

四字熟語の判定について

このプログラムで一番苦労したところ

《第１表》paintメソッドの処理内容

タイトルの表示
以下の処理を無限に繰り返す
ゲームメイン(すべて正解するか、ミスを３回するまで繰り返す)
選択文字数が４になるまで繰り返す
８×６のマスを作る
カーソルの移動
ソフトキー１なら文字消去，選択キーなら文字追加
選択文字の表示
正しい四字熟語かどうか
四字熟語の判定表示
正解発表(ここでソフトキー２を押すと，終了します)

は，コンピュータに四字熟語が正しいかを判定させる部分です。

選択した四字熟語が正しいかどうかは，四字熟語表(jyukugo[])にあるすべての(全部で450)熟語を調べます。そのため，正解した熟語が問題作成で選ばれた熟語と異なる場合があります。このような場合，全問正解ができないことがあります。

たとえば「行雲流水」「離合集散」「五里霧中」「意気消沈」が問題に選ばれていたときに，プレイヤーが「雲散霧消」を熟語に選んでも正解になります。

しかし，それら四つの熟語はそれぞれ１字ずつ足りなくなってしまうので，全問正解はまず不可能でしょう。

また，「左顧右眄」が出題されて「右顧左眄」を解答すると正解と判定しますが，正解発表の時には反転表示されない場合があります。

このことを問題として何らかの対策を考えることもありえますが，むしろこのほうがゲームとしてはおもしろいのではないかと思われました。

CHECKER FLAG

編：四字熟語判定に苦労したみたいですね。判定処理自体は単純に行えますが，漢字を一つずつ表示するというこのゲームの性質と，四字熟語そのものの性質に苦労させられたという感じです。
Dr.D：このプログラムでは，熟語の数そのものを増やすという方法で解決しておる。この他にも，ゲームシステムの工夫でさまざまな改善方法がありそうじゃ。
つぐ美：でも，こういった不具合に対処しているってのはテストプレイをしっかりしている証拠ですよね。
影：うう～，一つも知っている熟語が見つからない。

《第２表》変数表

int x, y	カーソルの位置
int ix,iy	カーソルの速度
boolean se,sk1,sk2	決定キー，ソフトキー１が押されたか
String fig	選択熟語の文字列
int pt	正解熟語数
int mis	不正解熟語数
String jyukugo[]	四字熟語表
String jg[]	問題とする四字熟語
String kj[]	その四字熟語の漢字
int selnb[]	選択した漢字の場所
int kjo[]	マスにある漢字の場所

そこで私は，なるべく同じ文字が使われている四字熟語を，熟語表の熟語に選びました。

判定で避けなければならないのは，正しい四字熟語なのにコンピュータが誤りと判定する場合です。前にあげた，「行雲流水」～「意気消沈」の四つが熟語表にあって「雲散霧消」がない場合は，「雲散霧消」が誤りと判断されてしまいます。これを避けるために，四字熟語の数を増やしたのですが，それに伴ってゲームシステムは貧弱になってしまいました。

改造について

このゲームの評価は，どの四字熟語をマスの中に表示させるかということが影響します。このプログラムでは，ただ単に熟語表からランダムに選んでいるだけですが，もっとおもしろい方法があるかもしれません。

このプログラムのゲーム性を高めるならば，正しい四字熟語と読みが似ている漢字もマスの中に入れてしまうことが考えられます。和文コードは，基本的に漢字の音の順に並んでいるので，熟語に使われた漢字のJIS，シフトJIS，区点コードがわかれば，その周辺のコードを使えば簡単に作ることができます。また，Unicodeの場合は形が似たものが集まっているので，Unicodeを利用する方法も考えられます。しかし，残念ながら，私はN88-日本語BASIC(86)ではできても，ｉアプリでは，私の技術不足でできません。

《リスト１》Main.javaのプログラム・リスト

```
/*  四字熟語を探せ  Ver. 1.51 programmed by T.Tsukuda */
import com.nttdocomo.ui.*;
import java.util.*;

public class Main extends IApplication{
    public void start(){
        Screen screen = new Screen();
        Display.setCurrent(screen);
    }
}

class Screen extends Canvas{
```

勉強のために使うのであれば，その意味を理解していないと意味がないので，四字熟語の意味もプログラムに入れようと考えました。しかし，熟語の意味まで入れてしまうと実行ファイルの最大サイズ(503iシリーズの場合10kバイト)を超えてしまうので，やめてしまいました。

エミュレータの動作について

ｉアプリ開発で使用するエミュレータでも，ゲームをプレイすることができます。このプログラムは，DoJaのエミュレータで動作します(**写真４**)。

★参考にしたもの
・加藤道理ほか 編著「新編常用国語便覧」(浜島書店，1991)
・岩波書店辞書編集部 編「岩波四字熟語辞典」(岩波書店，2002)

★自作一代記で，LANのWalkON機能を使ってPCをタイマ起動させるPCIボードを作ることはできないでしょうか。(大阪市・**玉出のタマ 37歳**)……【**編：**出番です，松原先生。あれ？松原先生～。**影：**姿が見えないということは，すでに研究に入っているとか。うーん。】

```
static int x=0; static int y=0; //カーソルの位置
static int ix=0; static int iy=0;//カーソルの速度
static boolean se=false; //決定キーが押されたか
static boolean sk1=false;
static boolean sk2=false;
static String fig=""; //選択熟語の文字列
static int pt=0; //正解熟語数
static int mis=0; //不正解熟語数
final static String jyukugo[]={ //四字熟語表
    "曖昧模糊","悪事千里","悪戦苦闘","悪人正機","阿鼻叫喚","安心立命",
    "暗中模索","唯唯諾諾","意気軒昂","意気消沈","意気投合","意気揚揚",
    "異口同音","意志薄弱","以心伝心","一衣帯水","一言居士","一期一会",
    "一日三秋","一日千秋","一念発起","一部始終","一網打尽","一目瞭然",
    "一問一答","一蓮托生","一攫千金","一喜一憂","一気呵成","一騎当千",
    "一挙両得","一刻千金","一切合切","一瀉千里","一触即発","一所懸命",
    "一生懸命","一心同体","一心不乱","一世一代","一世風靡","一石二鳥",
    "一朝一夕","一長一短","一刀両断","一発必中","韋編三絶","意味深長",
    "因果応報","慇懃無礼","隠忍自重","有為転変","右往左往","右顧左眄",
    "有象無象","海千山千","紆余曲折","雲散霧消","栄枯盛衰","会者定離",
    "円転滑脱","傍目八目","温厚篤実","温故知新","厭離穢土","外柔内剛",
    "快刀乱麻","偕老同穴","臥薪嘗胆","佳人薄命","花鳥風月","隔靴掻痒",
    "我田引水","画竜点睛","夏炉冬扇","感慨無量","汗牛充棟","換骨奪胎",
    "冠婚葬祭","勧善懲悪","完全無欠","艱難辛苦","頑迷固陋","機会均等",
    "奇怪千万","危機一髪","奇奇怪怪","危急存亡","起死回生","旗幟鮮明",
    "起承転結","喜色満面","疑心暗鬼","奇想天外","気息奄奄","喜怒哀楽",
    "牛飲馬食","九死一生","旧態依然","急転直下","驚天動地","興味津津",
    "虚虚実実","曲学阿世","玉石混交","挙国一致","虚心坦懐","毀誉褒貶",
    "金科玉条","欣喜雀躍","緊褌一番","金枝玉葉","金城湯池","空前絶後",
    "空即是色","空理空論","苦心惨憺","九分九厘","君子豹変","群雄割拠",
    "軽挙妄動","鶏口牛後","経世済民","軽佻浮薄","鶏鳴狗盗","月下氷人",
    "牽強付会","乾坤一擲","捲土重来","堅忍不抜","権謀術数","行雲流水",
    "厚顔無恥","綱紀粛正","剛毅木訥","巧言令色","公序良俗","広大無辺",
    "荒唐無稽","公平無私","豪放磊落","公明正大","呉越同舟","狐疑逡巡",
    "国士無双","孤軍奮闘","古今東西","古今無双","虎視眈眈","古色蒼然",
    "故事来歴","誇大妄想","鼓腹撃壌","五分五分","孤立無援","五里霧中",
    "欣求浄土","言語道断","斎戒沐浴","才気煥発","再三再四","才子佳人",
    "才子多病","才色兼備","左顧右眄","三寒四温","残酷非道","山紫水明",
    "三者鼎立","三段論法","三拝九拝","三位一体","三面六臂","自画自賛",
    "自家撞着","時期尚早","色即是空","自給自足","四苦八苦","試行錯誤",
    "自業自得","自己嫌悪","自己顕示","自己矛盾","自作自演","時時刻刻",
    "子子孫孫","事実無根","獅子奮迅","自縄自縛","自然淘汰","時代錯誤",
    "舌先三寸","七転八起","七転八倒","質実剛健","実践躬行","十中八九",
    "疾風迅雷","疾風怒濤","自暴自棄","四方八方","四面楚歌","自問自答",
    "弱肉強食","縦横無碍","縦横無礙","縦横無尽","自由闊達","終始一貫",
    "周章狼狽","秋霜烈日","十人十色","自由奔放","周知徹底","主客転倒",
    "熟読玩味","熟慮断行","取捨選択","酒池肉林","首尾一貫","純真無垢",
    "醇風美俗","順風満帆","小国寡民","笑止千万","盛者必衰","生生流転",
    "正真正銘","常套手段","枝葉末節","生老病死","諸行無常","初志貫徹",
    "私利私欲","支離滅裂","四六時中","心機一転","神経衰弱","真実一路",
    "唇歯輔車","神出鬼没","信賞必罰","針小棒大","新進気鋭","深謀遠慮",
    "森羅万象","酔生夢死","晴耕雨読","生殺与奪","誠心誠意","正正堂堂",
    "青天白日","清廉潔白","責任転嫁","是是非非","雪月風花","切磋琢磨",
    "切歯扼腕","絶体絶命","是非善悪","浅学非才","千客万来","千軍万馬",
    "千載一遇","千差万別","千紫万紅","全身全霊","前人未到","先制攻撃",
    "戦戦兢兢","前代未聞","全知全能","前途多難","前途洋洋","前途遼遠",
    "千篇一律","千変万化","先憂後楽","創意工夫","創業守成","漱石枕流",
    "率先垂範","大器晩成","大義名分","大言壮語","大山鳴動","泰山北斗",
    "泰然自若","大胆不敵","大同小異","大同団結","多岐亡羊","多事多難",
    "他人行儀","単純明快","男尊女卑","単刀直入","談論風発","遅疑逡巡",
    "知行合一","魑魅魍魎","中途半端","朝三暮四","跳梁跋扈","朝令暮改",
    "直情径行","猪突猛進","沈思黙考","適材適所","徹頭徹尾","天衣無縫",
    "天涯孤独","天下一品","天下泰平","電光石火","天災地変","天神地祇",
    "天真爛漫","天地神明","輾転反側","天罰覿面","天佑神助","当意即妙",
    "同工異曲","道聴塗説","東奔西走","独立独歩","徒手空拳","駑馬十駕",
    "内柔外剛","内憂外患","難攻不落","南船北馬","二者択一","二束三文",
    "肉袒負荊","日進月歩","二人三脚","如是我聞","二律背反","二六時中",
    "拈華微笑","博引旁証","博学多才","白砂青松","薄志弱行","博覧強記",
    "馬耳東風","八面六臂","八方美人","波瀾万丈","罵詈雑言","半信半疑",
    "万世不易","美辞麗句","人身御供","悲憤慷慨","誹謗中傷","百聞一見",
    "百家争鳴","百花繚乱","百鬼夜行","百発百中","表裏一体","疲労困憊",
    "品行方正","風光明媚","風林火山","不易流行","複雑怪奇","複雑多岐",
    "不倶戴天","不言実行","夫唱婦随","不即不離","物物交換","不撓不屈",
    "不偏不党","不老長寿","不老不死","付和雷同","粉骨砕身","文人墨客",
    "文武両道","平穏無事","片言隻句","片言隻語","暴虎馮河","傍若無人",
    "旁若無人","茫然自失","捧腹絶倒","抱腹絶倒","本末転倒","末法思想",
    "満場一致","満身創痍","三日天下","未来永劫","無為自然","無我夢中",
    "無茶苦茶","無念無想","無味乾燥","無欲恬淡","無理難題","明鏡止水",
    "明窓浄机","明眸皓歯","滅茶苦茶","面会謝絶","面従腹背","面目躍如",
    "孟母三遷","門前雀羅","問答無用","唯一無二","唯我独尊","有言実行",
    "優柔不断","有職故実","有名無実","勇猛果敢","悠悠閑閑","悠悠自適",
    "油断大敵","余韻嫋嫋","用意周到","羊頭狗肉","余裕綽綽","利害得失",
    "力戦奮闘","離合集散","立身出世","理非曲直","流言蜚語","竜頭蛇尾",
    "粒粒辛苦","良妻賢母","良知良能","理路整然","臨機応変","老少不定",
    "老若男女","和気藹藹","和光同塵","和魂漢才","和魂洋才","和洋折衷"
}; //ここにない熟語は誤りとみなされます。

public void paint(Graphics g){
    Font font=Font.getDefaultFont();
    int ht=font.getHeight();
    Random rnd = new Random();
    //タイトル表示
    g.lock();
    String str="四字熟語探し"; int wd=font.stringWidth(str);
```

★マイコンソフトの広告やなにわ通信で,「DISPL TV」の概要がわかってきた。購入するか否かは,現在検討中です。(神奈川県相模原市・K.栗田20歳)……【**なにわ:**ぜひ買ってや,よろしゅうたのむわ。**影:**なんか言葉がへん!ニセモノ?**なにわ:**何,言うとるねん。】

```
g.setColor(Graphics.getColorOfName(Graphics.WHITE));
g.fillRect(0,0, getWidth(), getHeight());
g.setColor(Graphics.getColorOfName(Graphics.BLACK));
g.drawString(str,(getWidth()-wd)/2,(getHeight()+ht)/2-3*ht-1);
str="［選択］キーを押してください";
for(int i=0;i<=font.stringWidth(str)/getWidth();i++){
    int j=font.stringWidth(str)/getWidth()+1;
    int k=font.stringWidth(str)/j;
    g.drawString(str.substring(i*str.length()/j,(i+1)*str.length()/j),
        (getWidth()-k)/2,(getHeight()+ht)/2+(i-1)*ht-1);
}
str="programmed by";
g.drawString(str,getWidth()-font.stringWidth(str),
    (getHeight()+ht)/2+2*ht-1);
str="T.Tsukuda";
g.drawString(str,getWidth()-font.stringWidth(str)
    ,(getHeight()+ht)/2+3*ht-1);
g.unlock(true);
while(!se){}
while(true){
    String jg[]=new String[12]; //問題とする四字熟語
    boolean jgox[]=new boolean[12];
    for(int i=0;i<12;i++){jgox[i]=false;}
    String kj[]=new String[48]; //その四字熟語の漢字
    int selnb[]=new int[4]; //選択した漢字の場所
    int kjo[] =new int[48]; //マスにある漢字の場所
    for(int i=0;i<48;i++){kjo[i]=i;}
    //場所のシャッフル
    for(int i=200;i>=0;i--){
        int s1=(rnd.nextInt()>>>1)%48;int s2=(rnd.nextInt()>>>1)%48;
        int j=kjo[s1]; kjo[s1]=kjo[s2]; kjo[s2]=j;}
    //問題とする熟語の選択
    for(int i=0;i<12;i++){
        jg[i]=jyukugo[(rnd.nextInt()>>>1)%jyukugo.length];
    }
    //各場所に漢字を当てはめる
    int w=(getWidth()/8-wd/6)/2+1;
    int h=(getHeight()/7-ht)/2+2;
    for(int i=0;i<12;i++){for(int j=0;j<4;j++){
        kj[4*i+j]=jg[i].substring(j,j+1);
    }}
    setSoftLabel(SOFT_KEY_1,"消去");
    setSoftLabel(SOFT_KEY_2,"降参");
    x=0;y=0;ix=0;iy=0;
    //ゲームメイン
    pt=0; mis=0; sk2=false; while(pt<12 && mis<3 && !sk2){
        fig="";
```

```
        while(fig.length()<4 && !sk2){
            g.lock();
            g.setColor(Graphics.getColorOfName(Graphics.WHITE));
            g.fillRect(0,0, getWidth(), getHeight());
            //8×6のマスを作る
            for(int i=0;i<48;i++){
                g.setColor(Graphics.getColorOfName(Graphics.BLACK));
                g.drawRect((i%8)*getWidth()/8,(i/8)*getHeight()/7+ht,
                    getWidth()/8, getHeight()/7);
                g.drawString(kj[kjo[i]], (i%8)*getWidth()/8+w,
                    (i/8+1)*getHeight()/7+ht-h);
            }
            //カーソルの表示
            g.setColor(Graphics.getColorOfName(Graphics.RED));
            g.fillRect(x*getWidth()/8,y*getHeight()/7+ht,
                getWidth()/8, getHeight()/7);
            g.setColor(Graphics.getColorOfName(Graphics.BLACK));
            g.drawRect(x*getWidth()/8,y*getHeight()/7+ht,
                getWidth()/8, getHeight()/7);
            g.drawString(kj[kjo[8*y+x]], x*getWidth()/8+w,
                (y+1)*getHeight()/7+ht-h);
            //選択文字の表示
            g.setColor(Graphics.getColorOfName(Graphics.WHITE));
            g.fillRect(0,0, getWidth(), ht);
            g.setColor(Graphics.getColorOfName(Graphics.BLACK));
            g.drawString(fig,
                (getWidth()-font.stringWidth(fig))/2,ht-1);
            g.unlock(true);
            //カーソルの移動
            se=false; while(!se && !sk1 && !sk2){
                while(ix==0 && iy==0 && !se && !sk1 && !sk2){}
                g.lock();
                g.setColor(Graphics.getColorOfName(Graphics.WHITE));
                g.fillRect(x*getWidth()/8,y*getHeight()/7+ht,
                    getWidth()/8, getHeight()/7);
                g.setColor(Graphics.getColorOfName(Graphics.BLACK));
                g.drawRect(x*getWidth()/8,y*getHeight()/7+ht,
                    getWidth()/8, getHeight()/7);
                g.drawString(kj[kjo[8*y+x]], x*getWidth()/8+w,
                    (y+1)*getHeight()/7+ht-h);
                x=(x+ix+8)%8; y=(y+iy+6)%6; ix=0; iy=0;
                g.setColor(Graphics.getColorOfName(Graphics.RED));
                g.fillRect(x*getWidth()/8,y*getHeight()/7+ht,
                    getWidth()/8, getHeight()/7);
                g.setColor(Graphics.getColorOfName(Graphics.BLACK));
                g.drawRect(x*getWidth()/8,y*getHeight()/7+ht,
                    getWidth()/8, getHeight()/7);
```

★本誌タイトルロゴや，投稿プログラムコーナーのプログラム賞など，少しずつ変わろうとしているベーマガの姿勢が感じられます。（埼玉県行田市・ジョアンナ21歳）……**【なにわ：**ベーマガも応援してやってや。**編：**なにわさんもご推薦，ベーマガをよろしゅう。**】**

```
            g.drawString(kj[kjo[8*y+x]], x*getWidth()/8+w,
                (y+1)*getHeight()/7+ht-h);
            g.unlock(true);
        }
        //ソフトキー１なら文字消去、選択キーなら文字追加
        if(sk1){sk1=false; fig="";
        }else{
            boolean hit=true; int t=0;
            while(t<fig.length() && hit){
                if(kjo[8*y+x]==selnb[t]){hit=false;}else{t++;}
            }
            if(hit && kj[kjo[8*y+x]].equals("×")==false){
                selnb[fig.length()]=kjo[8*y+x];
                fig=fig+kj[kjo[8*y+x]];
    }}}
    //正しい四字熟語かどうか
    boolean hit=false; int t=0; while(t<jyukugo.length && !hit){
        if(fig.equals(jyukugo[t])){hit=true;}else{t++;}
    }
    //四字熟語の判定表示
    g.lock();
    g.setColor(Graphics.getColorOfName(Graphics.WHITE));
    g.fillRect(0,0, getWidth(), ht);
    g.setColor(Graphics.getColorOfName(Graphics.RED));
    if(hit){
        g.drawString("○  "+fig,(getWidth()-wd)/2, ht-1);
        pt++;
        t=0; while(t<12){
            boolean j=false;
            if(fig.equals(jg[t]) && !jgox[t] ){
                jgox[t]=true; t=11;
            }
            t++;
        }
        for(int i=0;i<4;i++){kj[selnb[i]]="×";}
    }
    else if(!sk2){
        g.drawString("×  "+fig,(getWidth()-wd)/2, ht-1); mis++;
    }
    g.unlock(true);
    se=false; while(ix==0 && iy==0 && !se && !sk2){}
}
//正解発表
setSoftLabel(SOFT_KEY_1,"再開");
setSoftLabel(SOFT_KEY_2,"終了");
g.lock();
g.setColor(Graphics.getColorOfName(Graphics.WHITE));
```

```
        g.fillRect(0,0, getWidth(), getHeight());
        g.setColor(Graphics.getColorOfName(Graphics.BLACK));
        g.drawString("  正解発表  ",(getWidth()-wd)/2, ht-1);
        for(int i=0;i<12;i++){
            if(jgox[i]){
                g.setColor(Graphics.getColorOfName(Graphics.RED));
                g.fillRect((i%2)*getWidth()/2,(i/2)*getHeight()/7+ht,
                    getWidth()/2-1, getHeight()/7);
                g.setColor(Graphics.getColorOfName(Graphics.BLACK));
            }
            g.drawRect((i%2)*getWidth()/2,(i/2)*getHeight()/7+ht,
                getWidth()/2-1, getHeight()/7);
            g.drawString(jg[i],
                (i%2)*getWidth()/2+(getWidth()/2-wd*2/3)/2+1,
                (i/2+1)*getHeight()/7+ht-h);
        }
        g.unlock(true);
        se=false; sk1=false; sk2=false; while(!se && !sk1){
            if(sk2){IApplication.getCurrentApp().terminate();}
    }}}

    /*  キー入力処理  */

    public synchronized void processEvent(int type, int param) {
        if (type==Display.KEY_PRESSED_EVENT) {
            if (param==Display.KEY_SELECT) {se=true;}
            if (param==Display.KEY_LEFT  ) {ix--;}
            if (param==Display.KEY_RIGHT ) {ix++;}
            if (param==Display.KEY_UP    ) {iy--;}
            if (param==Display.KEY_DOWN  ) {iy++;}
            if (param==Display.KEY_SOFT1 ) {sk1=true;}
            if (param==Display.KEY_SOFT2 ) {sk2=true;}
}}}
```

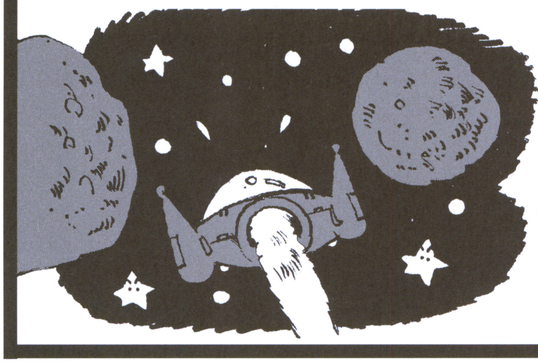
《リスト２》jyukugo.jamのプログラム・リスト

```
PackageURL = jyukugo.jar
AppSize = 9991
AppName = 四字熟語探し
AppVer = 1.51
AppParam =
AppClass = Main
SPsize =
LastModified = Sun, 20 Oct 2002 14:10:58
KvmVer =
```


★（1月号53ページのOFより）「P/ECE HAND BOOK Vol.2」で勉強して，PIECE作品を投稿させていただきます。（神奈川県小田原市・**大貧民17歳**）……**【影：**「P/ECE HAND BOOK Vol.2」好評発売中！ぜひ買って勉強して，飽きたら私にPIECEちょうだい。**編：**うん？**】**

WonderWitch用

小惑星を避ける・破壊する
Asteroid（アステロイド）



江口　健太郎（27歳）

はじめに

WonderWitchで，ほぼ初めて作ったプログラムです，スプライトを動かすテストプログラムに，いろいろと付け足していってできた感じです。

ちなみに，ワンダースワンカラー専用です。

遊び方

起動すると，タイトル(**写真１**)とスコアランキングが一定間隔で交互に表示されます。

Aボタンを押すとスタートします。Xキーで自機が動き，Aボタンで弾が発射されます(**写真３**)。進行するだけで得点が加算されます。また，小惑星を破壊しても得点が加算されます。

自機と小惑星が接触するとゲームオーバーです。

得点がベスト５に入ったら，名前を三文字で入力してください(**写真４**)。Xキーの左右で点滅してる文字を変更できます。Aで決定，Bで後退します。

入力方法

テキストエディタでasteroid.c(**リスト１**)とasteroid.cf(**リスト２**)を入力し，「lcc86 -o asteroid.bin asteroid.c」でコンパイルしてbinファイルを作成し，「mkfent asteroid.cf」でasteroid.fxファイルを出力し，それをTransMagicでWonderWitchカートリッジへ転送してください。

chara.bmpの画像

fonts.bmpの画像

explosion.bmpの画像

画像データについて

リストには画像データは含まれていません。Windows付属の「ペイント」などで，「chars.bmp」(64×8)，「fonts.bmp」(128×48)，「explosion.bmp」(16×16)の三つの画像ファイルを作成し，16色モードで保存してください。

これらのファイルを「bmpcnv -m3 ファイル名」コマンドでfrファイルに変換し，TransMagicでWonderWitchカートリッジへ転送してください。

プログラムについて

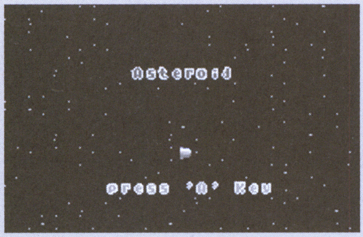
メインループの中でなるべく関数を呼びたくなかったので，ループに入る前に乱数表を作り，それを参照しています。

以前は，メインループ中でrand()関数を呼んでいましたが速度が遅いので，このような方式に変えました。

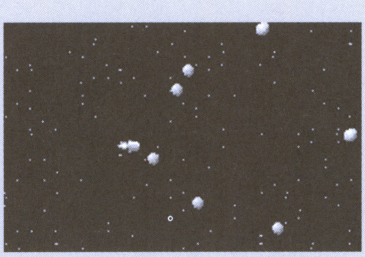
常に動作速度が一定に近い様にするために，画面に出現していない小惑星，弾にも当たり判定などをさせています。

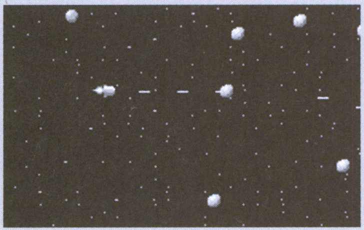
改造

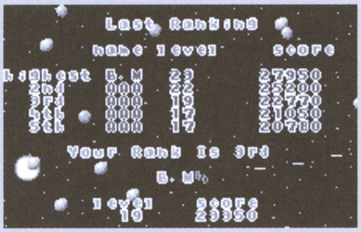
#defineの定数をいじって難易度を調



《写真１》スプライト機能を利用したゲーム


《写真２》迫り来る小惑星をかわし続ける


《写真３》小惑星は，ショットで破壊も可能



《写真４》得点ランキングでハイスコアを保存

★最近，ベーマガが届く→ハガキを書こうと思う→…が，気がつくとすでに次号が届いている→今月こそ書くぞ～と思う→…以下略。という感じです。ということで，プレゼントありがとうございました…いつの話だよ。(島根県美濃郡・K25歳) ……【影：おい，おい。】

《第１表》変数表

print_str	nullまでの文字列を表示します。
print_num	数値を表示します。end_xが最下位桁の表示される位置です。前に表示された数字を消したりはしないので，例えば20を表示した後，同じend_xの値で1を表示しようとすると21に見えます。
palette_setter	intの配列から16色分カラーをセットします。
print_ranking	スクリーン1か2に，スコアランキングベスト5を表示します。
draw_space	一つのフォントデータを上下左右にランダムに反転させて，背景の星を描きます。
initialize	パレット，フォント，スプライト，各オブジェクトの初期化を行います。
title	タイトルとスコアベスト5を表示します。
main_loop	メインループです。
ending	得点表示，ベスト5にスコアが入れば，名前入力。同点だった時は，先にその点数を取った方が上位になります。
main	メイン関数。

節することができます。asteroid_max_numは，40を超えると動作速度が目に見えて落ちます。

小惑星が破壊される時にアニメーションが欲しいですが，全般的にアニメーションすることをまったく考えてなかったので…。また，音も欲しいところです。

★参考にしたもの

・マイコンBASICマガジン　2001年９月号

・ワンダースワンゲームプログラミング(長久勝 著　ソフトバンクパブリッシング株式会社)

CHECKER FLAG

つぐ美：シューティングというよりは障害物避けゲームですが，動きがスムーズなのでプレイしていて気持ちいいですよ。

影：前移動と後ろ移動では移動スピードが違うなど，細かな演出がなされている。また，ハイスコア画面などの構成もしっかりしているゾ。

Dr.D：また，リストの組み方にも注目してほしい。特に説明文にもあるように，動作速度の面においては，乱数表の作成や当たり判定のやり方など，さまざまな工夫がなされておるリストじゃゾ。

```
    int level;                /* 到達レベル       */
    long score;               /* スコア */
}scrs;

typedef struct{               /* ゲーム結果構造体 */
    int achievement;          /* 到達レベル       */
    int destroy;              /* 破壊した数       */
    int state;                /* startで抜けた,激突,クリア */
}result;

int ssc_star=0x1003;          /* screen_set_char() に渡す背景(星)データ       */
int ssc_space=0x1004;         /* screen_set_char() に渡す背景(何も無し)データ */
```

《リスト１》asteroif.cのプログラム・リスト

```
#include <sys/bios.h>
#include <sys/libwwc.h>
#include <stdlib.h>
#include <stdio.h>
#include <sys/fcntl.h>
#include <resource.h>

#define asteroid_max_num 18      /* 最大小惑星数　オブジェクトの作られる数 */
#define shell_num 5              /* 弾の数　オブジェクトの作られる数       */
#define asteroid_hit_point 3     /* 小惑星の耐久度                         */
#define asteroid_hiding_x 225    /* 小惑星の画面外で待機するX座標          */
#define asteroid_hiding_y 145    /* 小惑星の画面外で待機するY座標          */
#define shell_hiding_x 8         /* 弾の画面外で待機するX座標              */
#define shell_hiding_y 200       /* 弾の画面外で待機するY座標              */
#define shell_speed 4            /* 弾の移動速度                           */
#define shoot_blank_value 5      /* 弾を撃つ間隔                           */
#define fsn 100                  /* 文字フォントの始まる位置               */
#define peak_level 30            /* mainのasteroid_now_max_numがこの数値に達すると
                                    asteroid_now_max_numの減少が始まる。   */
#define up_rate 100              /* asteroid_now_max_numが増減するペース   */
#define rnd_waiting 60           /* 小惑星が消えてから出現するまでの待機期間の範囲 */
#define neg_vec -4
#define ranking_fname "/ram0/ranking.log"    /* スコアログファイル名 */

#define cls(sc) screen_fill_char(sc,0,0,32,32,0x1004)  /* '何も無し'fontで埋める */

typedef struct{            /* 小惑星構造体 */
        int x;             /* X座標        */
        int y;             /* Y座標        */
        int vx;            /* 横移動速度   */
        int vy;            /* 縦移動速度   */
        int inc_y;         /* 縦移動率     */
        int inc_y_stock;/* inc_yを溜める*/
        int waiting;       /* 待機期間     */
        int hit_point;     /* 耐久度       */
}asteroid;

typedef struct{            /* 弾構造体 */
         int x;            /* X座標    */
         int y;            /* Y座標    */
         int vx;           /* 弾速度   */
}shell;

typedef struct{            /* スコア構造体      */
    char name[4];          /* 名前。3文字+NULL */
```

★ウチのクラスの情報処理の授業，パソコンがよく止まります。多分，影さんだと思うので，つぐ美さん，よろしくお願いします。／カップヌ○ドルの宇宙人のフィギュア，全部集まりませんでした。（東京都練馬区・**ルストフレム15歳**）……【**影：それからどした。**】

```c
unsigned char explosion_data[]={   /* sprite_set_data() に渡す自機爆発データ */
        0x08,0x00,0xe8,0xe8,
        0x09,0x00,0xe8,0xe8,
        0x0a,0x00,0xe8,0xe8,
        0x0b,0x00,0xe8,0xe8,
};
int eighth_colors[]={             /* パレットナンバー８番 */
    0x0000,0x0765,0x0000,0x0feb, /* 未使用,暗茶  ,未使用,明明茶*/
    0x0000,0x0ba7,0x0ff9,0x0bbb, /* 不使用,  茶  ,  弾  ,明灰  */
    0x0777,0x0f33,0x0000,0x0dc9, /* 暗灰  ,  赤  ,未使用,明茶  */
    0x077f,0x0ccf,0x0eee,0x0fff  /* 青    ,青白  ,明明灰,  白  */
};
char *rank_str[]={    /* スコアランク表示に使う文字列 */
    "highest","2nd","3rd","4th","5th"
};

FILE far *fp;         /* スコアログの為のファイルポインタ             */
char far *fonts;      /* mmap() で得るアドレスを受けるポインタ変数    */
int waiting_rnd_tbl[256];   /* 小惑星待機期間用乱数表     */
int y_aprc_rnd_tbl[256];    /*       Ｙ出現位置用乱数表   */
int x_aprc_rnd_tbl[256];    /*       Ｘ出現位置用乱数表   */
int switch_rnd_tbl[256];    /*       出現位置決定用乱数表 */
int vx_rnd_tbl[256];        /*       速度用乱数表         */
int angle_rnd_tbl[256];     /*       角度用乱数表         */
int vy_rnd_tbl[256];        /*       上か下か乱数表       */
char nm_tmp[6];             /* 名前入力に使用          */
asteroid asteroids[asteroid_max_num];  /* 小惑星構造体オブジェクト */
shell shells[shell_num];               /* 弾構造体オブジェクト     */
scrs scores[6];                   /* スコア構造体オブジェクト６個。６個目は新規用 */
/***** 文字列表示関数 *****/
void print_str(char screen,char x,char y,char* str,int font_start_num,int
pallete_num){
    static int bmd; /* bit map data : screen_set_charに渡す */
    unsigned int i;    /* index */
    for(bmd=pallete_num<<9,i=0;str[i];bmd=pallete_num<<9,i++){
        bmd=bmd+font_start_num+str[i]-' ';
         screen_set_char(screen,x+i,y,1,1,&bmd);
     }
}
/***** 数値表示関数 *****/
void print_num(char screen,char end_x,char y,long num,int font_start_num,int
pallete_num){

    static int bmd;    /* bit map data : screen_set_char()に渡す */
    unsigned int i;    /* index */
    for(bmd=pallete_num<<9,i=0  ;num/10  ;bmd=pallete_num<<9,i--,num/=10){
            bmd=bmd+font_start_num+'0'-' '+num%10;
             screen_set_char(screen,end_x+i,y,1,1,&bmd);
        }
        bmd=bmd+font_start_num+'0'-' '+num%10;
        screen_set_char(screen,end_x+i,y,1,1,&bmd);
}
/***** パレット設定関数 *****/
void palette_setter(int pallete_num,int colors[]){
    int i;
    for(i=1;i<16;i++){
         wwc_palette_set_color( pallete_num,i,colors[i]);
    }
}
/***** ランキング表示関数 *****/
void print_ranking(int scr){ /* screen scr にランキングを表示する。*/
    int i;
     print_str(scr,7,3,"name",fsn,0);
     print_str(scr,12,3,"level",fsn,0);
     print_str(scr,21,3,"score",fsn,0);

     for(i=0;i<5;i++){
         print_str(scr,(i)? 2:0 ,5+i,rank_str[i],fsn,0);
         print_str(scr,8,5+i,scores[i].name,fsn,1);
         print_num(scr,14,5+i,scores[i].level,fsn,1);
         print_num(scr,24,5+i,scores[i].score,fsn,1);
     }
}
/***** 背景描画関数 *****/
void draw_space(){    /* 一つのフォントをランダムで縦横反転させて描く */
    int i,j,rd,mask;
    cls(0);
    for(i=0;i<32;i++){
          for(j=0;j<32;j++){
               if((rd=rand()%10)<3){
                   switch(rd){
                          case 0: mask=0x8000; break;
                          case 1: mask=0x4000; break;
                          case 2: mask=0xc000; break;
                   }
                   ssc_star=ssc_star^mask;
                   screen_set_char(0,j,i,1,1,&ssc_star);
               }else{
                    screen_set_char(0,j,i,1,1,&ssc_space);
               }
          }
     }
```

★祝・ペグちゃん復活！今後の活躍に期待しています。（東京都八王子市・アーデルハイド20歳）……【**編：**これからどんどん活躍してもらいましょう。**影：**ま，ライバルは多いほうが私の活力にもなるからね。**つぐ美：**ペグちゃんをライバル視してどうすんの！】

```
}
/***** 初期化 *****/
void initialize(){
    int i;
    srand(rtc_get_datetime(6));        /* 乱数の種 */
    wwc_set_color_mode(COLOR_MODE_16PACKED);
    palette_setter(8,eighth_colors);
    wwc_palette_set_color(0,0,0x0000);
    wwc_palette_set_color(0,1,0x0e33); /*文字用  赤*/
    wwc_palette_set_color(0,15,0xfff);
    wwc_palette_set_color(1,0,0x0000);
    wwc_palette_set_color(1,1,0x044f); /* 文字用 青*/
    wwc_palette_set_color(1,15,0xfff);
    wwc_palette_set_color(2,0,0x0000);
    wwc_palette_set_color(2,1,0x00c0); /*文字用  緑 */
    wwc_palette_set_color(2,15,0xfff);
    display_control(0x0005);

    /* フォントデータをセット */
    fonts=mmap("/rom0/chars.fr");        /* 自機、噴炎、弾、小惑星 */
    wwc_font_set_colordata(0,8,fonts+sizeof(res_image_t)+32);
    fonts=mmap("/rom0/fonts.fr");        /* 文字 */
    wwc_font_set_colordata(fsn,96,fonts+sizeof(res_image_t)+32);
    fonts=mmap("/rom0/explosion.fr");    /* 爆発 */
    wwc_font_set_colordata(8,4,fonts+sizeof(res_image_t)+32);

    /* スプライトの設定 */
    sprite_set_range(0,2+asteroid_max_num+shell_num+4); /* 使用スプライト個数 */
    sprite_set_char_location(0,0x0000,110,90);   /* 自機 */
    sprite_set_char_location(1,0x0001,0,145);    /* 噴炎 */
    for(i=0;i<asteroid_max_num;i++){        /* 小惑星 */
        sprite_set_char_location(2+i,0x0005+rand()%3,
            asteroid_hiding_x,asteroid_hiding_y);
    }
    for(i=0;i<shell_num;i++){                /* 弾 */
         sprite_set_char_location(2+asteroid_max_num+i,0x0002,
             shell_hiding_x,shell_hiding_y);
    }
    sprite_set_data(2+asteroid_max_num+shell_num,4,explosion_data); /* 爆発 */

    for(i=0;i<asteroid_max_num;i++){    /* asteroid 初期化 */
        asteroids[i].waiting=rand()%300;
        asteroids[i].x=asteroid_hiding_x;
        asteroids[i].y=asteroid_hiding_y;
        asteroids[i].vx=0;
        asteroids[i].vy=0;
        asteroids[i].inc_y=rand()%11;
        asteroids[i].inc_y_stock=0;
        asteroids[i].hit_point=asteroid_hit_point;
    }

    for(i=0;i<shell_num;i++){            /* shell 初期化 */
        shells[i].x=shell_hiding_x;
        shells[i].y=shell_hiding_y;
        shells[i].vx=0;
    }
}
/***** タイトル画面 *****/
void title(){
    int flg=2;        /* どちらを表示するかのフラグ */
    unsigned int count=1; /* 65535 の次にゼロになる */
   memset(scores,0x00,sizeof(scrs)*6);    /* ゼロで埋める */
    fp=fopen(ranking_fname,"r");
    if(fp!=NULL){    /* スコアログファイルが存在する時 */
        fread(scores,sizeof(scrs),5,fp);
        fclose(fp);
    }
   while(flg){
        flg=flg%2+1;    /* 1なら2に、2なら1に。*/
        draw_space();
        switch(flg){
            case 1:
              print_str(0,10,5,"Asteroid",fsn,1);
              print_str(0,8,14,"press 'A' Key",fsn,2);
              break;
          case 2:
              print_str(0,10,1,"Ranking",fsn,2);
              print_ranking(0);
              break;
        }
        while(count++!=0)    if(key_press_check() & KEY_A) { flg=0;break; }
    }
}
/***** メインループ 関数 *****/
void main_loop(result far *r_obj_p){
    int r;          /* key_press_check()等の返値を受ける */
    int i,j;        /* forループで使用 */
    int x=80,y=70;        /* 自機座標 */
    int shoot_blank=0;    /* 弾を撃てる間隔 */
    int shot=0;           /* 発射ボタンを押し、且つ間隔が十分な時1になる */
    int asteroid_now_max_num=1;    /* 現在の小惑星最大出現数    */
                 /* 又、この値は進行度を表し、asteroid_max_numを超える。*/
    int asteroid_now_num=0;      /* 小惑星の現在の出現数       */
    int destroy=0;               /* 破壊した小惑星の数        */
```

★ADSLなので，WindowsXPのSP1やIEの修正パッチが気軽に入手できるけど，そのパッチのインストール作業に時間取られてます。（神奈川県川崎市・**浦辺25歳**）……【**影**：あとはPCのパフォーマンスを上げることだね。**編**：そういう影さんもHDの容量アップしたら？】

```c
int vector=1;                /* 小惑星の数が増加するか減少するか */
int end_time=up_rate*peak_level/neg_vec*-1+229+100;
                    /* 小惑星の数の減少開始からの終了までの時間 */
unsigned int sx=0;  /* スクロール時これをインクリメントする */
unsigned char waiting_rnd_idx=0;    /* 乱数表用インデックス        */
unsigned char y_aprc_rnd_idx=0;     /* 表の大きさは256で           */
unsigned char x_aprc_rnd_idx=0;     /* これらの変数を++していくと */
unsigned char sw_rnd_idx=0;         /* 255の次は0になるので       */
unsigned char vx_rnd_idx=0;         /* 境界判定はしない            */
unsigned char agl_rnd_idx=0;
unsigned char vy_rnd_idx=0;
char flg=1;                         /* 自機の噴炎のゆらぎの表現に使う */

for(i=0;i<256;i++){                 /* 乱数表作成 */
    waiting_rnd_tbl[i]=rand()%rnd_waiting+1;
    y_aprc_rnd_tbl[i]=rand()%145;
    x_aprc_rnd_tbl[i]=rand()%54+170;
    switch_rnd_tbl[i]=rand()%5;
    vx_rnd_tbl[i]=-(rand()%4+1);
    angle_rnd_tbl[i]=rand()%31;
    vy_rnd_tbl[i]=rand()%3-1;
}

while(1){
    r=key_press_check();
    if(r & KEY_X4 && x>0  ) x--;
    if(r & KEY_X2 && x<216) x++;
    if(r & KEY_X1 && y>2  ) y--;
    if(r & KEY_X3 && y<136) y++;
    if(r & KEY_A) if(!shoot_blank) shot=1;  /* shoot_blankが0なら撃てる */
              else   shoot_blank--;        /* 0でないなら待つことになる */

    if(r & KEY_START) {
        (*r_obj_p).state=0;
        return;        /* startを押したら抜ける */
    }

    sprite_set_location(0,x,y); /*自機*/

    if(!flg){
        sprite_set_location(1,x-8,y); /*  炎 */
        flg=3;        /*  ループ実行３回に１回だけ位置を変える */
    }
    flg--;

    for(i=0;i<asteroid_max_num;i++){    /* 各小惑星の動作 */
        sprite_set_location(2+i,asteroids[i].x,asteroids[i].y);

        if(asteroids[i].waiting==0 ){    /* 待機終了 */
            if( asteroid_now_num<asteroid_now_max_num ){
                asteroid_now_num++;    /* 数が現在の最大出現数未満なら出現 */
                switch(    switch_rnd_tbl[sw_rnd_idx++] ){
                    case 0:
                        asteroids[i].x=x_aprc_rnd_tbl[x_aprc_rnd_idx++];
                        asteroids[i].vx=vx_rnd_tbl[vx_rnd_idx++];
                        asteroids[i].vy=1;
                        asteroids[i].y=0;
                        asteroids[i].inc_y=angle_rnd_tbl[agl_rnd_idx++];
                        break;
                    case 1:
                    case 2:
                    case 3:
                        asteroids[i].x=225;
                        asteroids[i].vx=vx_rnd_tbl[vx_rnd_idx++];
                        asteroids[i].y=y_aprc_rnd_tbl[y_aprc_rnd_idx++];
                        asteroids[i].vy=vy_rnd_tbl[vy_rnd_idx++];
                        asteroids[i].inc_y=angle_rnd_tbl[agl_rnd_idx++];
                        break;
                    case 4:
                        asteroids[i].x=x_aprc_rnd_tbl[x_aprc_rnd_idx++];
                        asteroids[i].vx=vx_rnd_tbl[vx_rnd_idx++];
                        asteroids[i].vy=-1;
                        asteroids[i].y=144;
                        asteroids[i].inc_y=angle_rnd_tbl[agl_rnd_idx++];
                        break;
                    default:    ;
                }
                asteroids[i].waiting=-1;
                asteroids[i].hit_point=asteroid_hit_point;
            }else{    /* 出現数制限で出られないときは、再度待つ */
                 asteroids[i].waiting=waiting_rnd_tbl[waiting_rnd_idx++];
            }
        }/* 以上、待機終了時 */

        if(x>asteroids[i].x-5 && x<asteroids[i].x+5 /* 自機と衝突した時 */
            && y>asteroids[i].y-5 && y<asteroids[i].y+5 ){
                (*r_obj_p).state=1;
                goto label;
        }

        if(asteroids[i].x<-4 || asteroids[i].y>asteroid_hiding_y
            || asteroids[i].y<-3){
```


★「う˝っ…バ，バグゥ〜」「グバァー」「バ…バグゥ〜」「…話が終わらないじゃない！」「グ…バグゥ〜」（愛知県名古屋市・ズオウ・リアキ30歳）……【影：うぐぅっ！ばぐっ！やられた，今日の仕事は打ち止めだ。編：何にやられた？影：私のPCがウイルスに〜！】

```
            /* 画面外に消えた時 */
             asteroids[i].waiting=waiting_rnd_tbl[waiting_rnd_idx++];
             asteroids[i].vx=0;
             asteroids[i].vy=0;
             asteroids[i].x=asteroid_hiding_x;
             asteroids[i].y=asteroid_hiding_y;    /* 145 */
             asteroid_now_num--;
        }

        /* inc_yの値によりyの増加のペースが決まる。*/
        asteroids[i].inc_y_stock+=asteroids[i].inc_y;
        if(asteroids[i].inc_y_stock>29){
            asteroids[i].y=asteroids[i].y+asteroids[i].vy;
            asteroids[i].inc_y_stock=0;
        }

        asteroids[i].x+=asteroids[i].vx;
        asteroids[i].waiting--;

    } /* 以上、小惑星のforループ */

    for(i=0;i<shell_num;i++){    /* 各弾の動作 */
        /* shotが１でこの弾がまだ発射されてないなら */
        if(shot && shells[i].vx==0){
             shells[i].y=y;
             shells[i].x=x+5;
             shells[i].vx=shell_speed;
             shoot_blank=shoot_blank_value;
             shot=0;
        }

        shells[i].x+=shells[i].vx;
        sprite_set_location(2+asteroid_max_num+i,shells[i].x,shells[i].y);

        for(j=0;j<asteroid_max_num;j++){/* 弾と小惑星の当たり判定 */
           if( shells[i].x>asteroids[j].x-7 && shells[i].x<asteroids[j].x+7
              && shells[i].y>asteroids[j].y-4 && shells[i].y<asteroids[j].y+4){
                      /* 当たり */
                      if(!asteroids[j].hit_point){ /* 小惑星耐久度ゼロ */
                      /* 共に画面外へ消えさせる */
                      asteroids[j].y=asteroid_hiding_y+1;
                          shells[i].x=226;
                          destroy++;
                      }else{    /* 耐久度ゼロでない */
                           asteroids[j].hit_point--;
                           shells[i].x=226;
                      }
                }/* 当たってもループ続行 */
            }    /* 以上、当たり判定forループ */

            if(shells[i].x>225){ /*画面外へ*/
                shells[i].x=shell_hiding_x;
                shells[i].y=shell_hiding_y;
                shells[i].vx=0;
            }
        } /* 以上、弾のforループ */

        screen_set_scroll(0,sx++,0);
        if(!(sx%up_rate) )    asteroid_now_max_num+=vector; /* 増加または減少 */
       if(asteroid_now_max_num>=peak_level) vector=neg_vec;
       if(vector<0 && !end_time--){ /* 左辺が偽ならend_time--は実行されない */
                     (*r_obj_p).state=2;break; /*end_timeが0で抜ける*/
       }

        sys_wait(0);
    }
    label:
        if((*r_obj_p).state==1){
            sprite_set_location(0,-16,-8);/*爆発の描画 */
            sprite_set_location(1,-16,-8);
            sprite_set_location(2+asteroid_max_num+shell_num,x-4,y-4);
            sprite_set_location(2+asteroid_max_num+shell_num+1,x+4,y-4);
            sprite_set_location(2+asteroid_max_num+shell_num+2,x-4,y+4);
            sprite_set_location(2+asteroid_max_num+shell_num+3,x+4,y+4);
        }
        if(vector==neg_vec)
                asteroid_now_max_num=peak_level+
                    (peak_level-asteroid_now_max_num)/vector*-1;
        (*r_obj_p).achievement=asteroid_now_max_num;
        (*r_obj_p).destroy=destroy;

}
/***** エンディング *****/
void ending(result far *p){
        int i,j,r,nm_index=0;
        int loop_flg=1;             /* ループフラグ */
        char flg;                   /* 点滅のフラグ */
        scrs sc_tmp;                /* スコア入れ換え時に使う */

        display_control(0x0007);    /* screen2 on */
        cls(1);
        scores[5].score=(*p).destroy*(100+(*p).achievement*10)+(*p).achievement*1000;
        scores[5].level=(*p).achievement;
```

★もうすぐ2003年の幕開けですなあ。早いもんです。私も20歳になってしまい，信じがたいです。まだまだ子供っぽいのに…。（青森県弘前市・**奥野健太郎19歳**）……【**影：**うちの編集部には異様に人生経験をつんだような20歳がいます。**つぐ美：**は？どこに？】

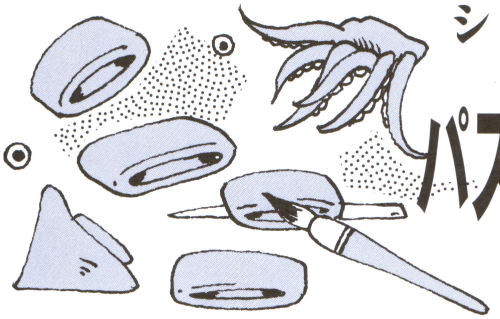
```
if((*p).state==2) print_str(1,10,7,"Cleared !",fsn,2);
else print_str(1,9,7,"Game Over...",fsn,1);
print_str(1,10,13,"score",fsn,1);
print_num(1,20,13,scores[5].score,fsn,0);
print_str(1,10,11,"level",fsn,1);
print_num(1,17,11,scores[5].level,fsn,0);

sys_wait(60);
key_wait();
cls(1);

print_str(1,8,1,"Last Ranking",fsn,2);
print_ranking(1);

for(i=5;i>0 && scores[i-1].score<scores[i].score;i--){ /*自順位を挿入*/
        memcpy(&sc_tmp,scores+i-1,sizeof(scrs));
        memcpy(scores+i-1,scores+i,sizeof(scrs));
        memcpy(scores+i,&sc_tmp,sizeof(scrs));
}

if(i>4) {    /* ベスト5に入れなかった時 */
    print_str(1,1,12,"You are out of ranking.",fsn,2);
    return;
}

print_str(1,5,11,"Your Rank Is",fsn,2);
print_str(1,18,11,rank_str[i],fsn,0);
print_str(1,7,15,"level    score",fsn,2);
print_num(1,10,16,scores[i].level,fsn,0);
print_num(1,19,16,scores[i].score,fsn,0);

/* 名前の入力 */
memset(scores[i].name,'A',3);    /* 最初は全てA */
memset(nm_tmp,0x00,6);
while(loop_flg){
    for(flg=0,j=0;j<6;j++){ /* jが0から5の間、flgが1になるのは */
        sys_wait(4);        /* 4から5。その間が点滅の滅になる  */
        if(j==4) flg=1;
        memcpy(nm_tmp,scores[i].name,4);
        nm_tmp[3]=nm_tmp[5];
        if(flg)    nm_tmp[nm_index]=' ';

        print_str(1,12,13,nm_tmp,fsn,2);

        r=key_hit_check_with_repeat();
         if(r & KEY_A )    nm_index++;
        if(r & KEY_B )    nm_index--;
        if(nm_index<0) nm_index=0;
        if(nm_index==4)    {loop_flg=0;break;}
        if(nm_index==3){    /*end mark表示*/
            nm_tmp[5]=0x7f;/* 7fはエンドマーク */
            continue;
        }else
            nm_tmp[5]=' ';/* エンドマークを消す為に空白表示 */

        r=key_press_check();
        if(r & KEY_X4 )    scores[i].name[nm_index]--;
        if(r & KEY_X2 ) scores[i].name[nm_index]++;

        if(scores[i].name[nm_index]<' ') scores[i].name[nm_index]='Z';
        if(scores[i].name[nm_index]>'Z') scores[i].name[nm_index]=' ';
    }
}
sys_wait(70);
cls(1);

print_str(1,9,1,"New Ranking",fsn,2);
print_ranking(1);

fp=fopen(ranking_fname,"w");
fwrite(scores,sizeof(scrs),5,fp);
fclose(fp);
}
/***** メイン関数 *****/
void main(){
    result r_obj;    /* ゲーム結果構造体オブジェクト */
    initialize();
    title();
    draw_space();
    main_loop(&r_obj);
    if(r_obj.state>0)    ending(&r_obj);
    key_wait();
}
```

《リスト2》asteroif.cfのプログラム・リスト

```
name: asteroid
info: asteroid
mode: x
source: asteroid.bin
output: asteroid.fx
```

★歌うぜ若人！レッツファイヤー！燃え上がれ，燃え上がれ，燃え上がれ！マクロス。愛・知冬はG！（愛知県名古屋市・**最馬場可？**歳）……【**Dr.D：**メラメラメラ！どうじゃ，この燃えよう。**影：**Dr.，書類が〜。**Dr.D：**むむっ，誰じゃこんなところでタバコを！】

ポケコン(FX-870P／890P)用



シャッフルされたキャラクタでパズル

パズルナイン for CASIO FX-870P

ゆーとんパパ　(33歳)

FX-890P

内容

1998年4月号に掲載されたVB用『ルービックスクウェア』(Yotchee改めよっちさん作)を参考に，ポケコンでパズルの画像をキャラクタにして作成してみました。

バラバラにシャッフルされた絵柄(縦3×横3)を，制限時間内に揃えるというパズルゲームです。ステージ数は全5面です。ステージが進むにつれて制限時間が少なくなり，難易度がアップします。

すべてのステージをクリアすると登場キャラが勢揃いしますので，がんばってみてください。

操作方法

プログラムを実行すると，2枚の絵柄が現れ，左側の絵柄がランダムにシャッフルされ並び換えられます。

右の絵柄を参考に，制限時間内に左側の絵柄を並べ直しすというパズルゲームです。

操作には，1～6キーを使用します。

1～3キーは，それぞれ上中下段の絵柄を右方向に1ブロック移動させ，一番右の絵柄を一番左に移動させます。

4～6キーは，それぞれ左中右列の絵柄を下方向に1ブロック移動させ，一番下の絵柄を一番上に移動させます。

各ステージをクリアすると，タイムボーナスとクリアボーナスがスコアに加算されます。

プログラムの仕組み

キャラクタは，ユーザー定義文字の4文字分を使用しています。

「FX-870Pでは4文字分までしかキャラクター定義できないのに，3×3の9文字分のキャラをどうやって表現しているのだろう？」と，思う人もいるかと思いますが，いたって簡単な仕組みです！

画面をスクロールなどで再描画しなければ，一度表示されたキャラは変わらないため，定義データを入れ換えて表示しているだけなのです(全面クリア時の画面表示では，45文字分もキャラクター定義に！…感動？)。

苦労した点

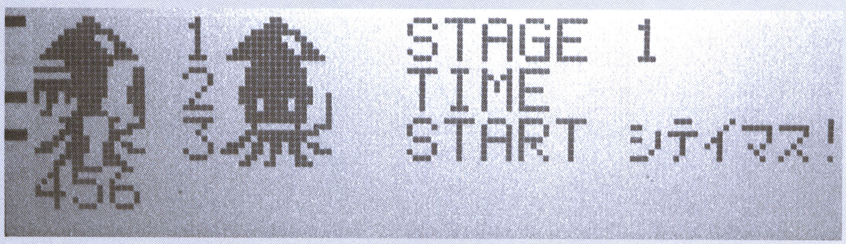
FX-870Pと890Pのどちらでも楽しめるように，共通コマンドのみを使用するようにしました。

タイムバーの表示が，キー操作をしている時と，そうでない時で減り方に差が生じるため，キー操作時のループ回数を調整し解消しました(200,320,370行のKの値)。

FX-890Pで楽しむ場合は，実行速度の違いがあるため，下記の行の数値を変更してください。

200行の「FOR K=1 TO 8」を「FOR K=1 TO 16」に変更。

320行と370行の「K=K+4」を「K=K+2」に変更。



《写真》左側のキャラクタをパネル移動で揃えていく

CHECKER FLAG

影：パズルの題材となるキャラクタが，非常にうまく描かれているね。

Dr.D：説明文にもあるように，ポケコンの画面表示の性質をうまく利用して，その個性あるキャラクタを表示している点が，このプログラムの工夫と言えるゾ。

編：ゲームとしては，制限時間がチョット厳しい印象ですね。時間内にクリアできないという人は，120行目で初期化される変数“TI”の値を，各自で調整してみるといいでしょう。

★1月号の93ページに触発されて投稿プログラムの感想を。12月号の「やじるしパズル ALAS!」は，画面に派手さはないけど，パズルとしての完成度が高い。(東京都葛飾区・影バイダー17歳)……【影：これには私もはまったぞ。みんなもぜひプレイしてみて。】

《リスト》パズルナインのプログラム・リスト

```
10 '------- ｼｮｷｾｯﾃｲ -------
20 CLEAR:RESTORE:SET N:DIM T$(7),A(9),A$(9):STN=5
30 FOR I=1 TO 7:READ A$:T$(I)=A$:NEXT
40 '------ ｽﾃｰｼﾞｼﾞｭﾝﾋﾞ ------
50 CLS:DEFCHR$(255)=T$(7):K=0
60 LOCATE 4,0:PRINT "1":LOCATE 4,1:PRINT "2":LOCATE 4,2:PRINT "3":LOCATE 0,3:PRINT "456";
70 FOR I=1 TO 9:A(I)=I:NEXT
80 FOR I=0 TO 2:FOR J=0 TO 2:K=K+1
90 READ A$(K):DEFCHR$(252)=A$(K)
100 LOCATE J,I:PRINT CHR$(252):LOCATE J+5,I:PRINT CHR$(252)
110 NEXT:NEXT
120 ST=ST+1:TI=34-ST*3
130 LOCATE 10,0:PRINT "STAGE";ST:LOCATE 10,1:PRINT "TIME"
140 FOR I=14 TO TI:LOCATE I,1:PRINT CHR$(255):NEXT:LOCATE 10,2:PRINT "ｼｬｯﾌﾙ ｼﾃｲﾏｽ!"
150 FOR I=1 TO 30
160 KE=INT(RAN#*10):IF KE<1 OR KE>6 THEN 160
170 GOSUB 260:NEXT
180 '------ ﾒｲﾝ ------
190 LOCATE 10,2:PRINT "START!      "
200 FOR I=TI TO 14 STEP-1:FOR J=6 TO 1 STEP-1:DEFCHR$(255)=T$(J):FOR K=1 TO 8
210 K$=INKEY$:KE=VAL(K$)
220 GOSUB 260:GOSUB 390
230 NEXT:LOCATE I,1:PRINT CHR$(255):NEXT:NEXT
240 LOCATE 10,1:PRINT "TIME IS OVER!"
250 LOCATE 10,2:PRINT "SCORE=";SC:A$="                          ":GOTO 600
260 IF KE>0 AND KE<4 THEN B=KE*3:A=A(B):A(B)=A(B-1):A(B-1)=A(B-2):A(B-
2)=A:A$=A$(B):A$(B)=A$(B-1):A$(B-1)=A$(B-2):A$(B-2)=A$:GOSUB 300
270 IF KE>3 AND KE<7 THEN A=A(KE+3):A(KE+3)=A(KE):A(KE)=A(KE-3):A(KE-
3)=A:A$=A$(KE+3):A$(KE+3)=A$(KE):A$(KE)=A$(KE-3):A$(KE-3)=A$:GOSUB 330
280 RETURN
290 '----- ﾊﾟｽﾞﾙﾋｮｳｼﾞ -----
300 DEFCHR$(252)=A$(B-2):DEFCHR$(253)=A$(B-1):DEFCHR$(254)=A$(B)
310 LOCATE 0,KE-1:PRINT CHR$(252);CHR$(253);CHR$(254)
320 K=K+4:RETURN
330 DEFCHR$(252)=A$(KE+3):DEFCHR$(253)=A$(KE):DEFCHR$(254)=A$(KE-3)
340 B=KE-4:LOCATE B,2:PRINT CHR$(252)
350 LOCATE B,1:PRINT CHR$(253)
360 LOCATE B,0:PRINT CHR$(254)
370 K=K+4:RETURN
380 '---- ｽﾃｰｼﾞｸﾘｱ ﾊﾝﾃｲ ----
390 FOR II=1 TO 9
400 IF A(II)<>II THEN RETURN
410 NEXT
420 '--- ｽｺｱｹｲｻﾝ & ﾋｮｳｼﾞ ---
430 LOCATE 18,0:PRINT "CLEAR!":GOSUB 530:LOCATE 10,2:PRINT "TIME BOUNS =":S=0:SB=ST^2*400
440 FOR II=I TO 14 STEP-1:FOR JJ=J TO 1 STEP-1:DEFCHR$(255)=T$(JJ)
450 LOCATE II,1:PRINT CHR$(255):S=S+ST*20:LOCATE 22,2:PRINT USING"#####";S
460 NEXT:J=6:NEXT
470 GOSUB 530:LOCATE 10,2:PRINT "CLEAR BOUNS =";SB:GOSUB 530
480 GOSUB 530:LOCATE 10,2:PRINT "STAGE SCORE =";S+SB:GOSUB 530
490 SC=SC+S+SB:LOCATE 10,2:PRINT "TOTAL SCORE =";SC
500 LOCATE 10,3:PRINT "HIT ANY KEY!";
510 K$=INKEY$:IF K$="" THEN 510
520 IF ST<STN THEN 50 ELSE 550
530 FOR II=1 TO 700:NEXT:RETURN
540 '--- ｵｰﾙｽﾃｰｼﾞｸﾘｱ ﾋｮｳｼﾞ ---
550 CLS:RESTORE 780:FOR K=1 TO 5:LOCATE 6*K-5,0:PRINTUSING "#.";K:FOR I=0 TO 2:FOR J=0 TO 2
560 READ A$:DEFCHR$(252)=A$
570 LOCATE 6*K-3+J,I:PRINT CHR$(252)
580 NEXT:NEXT:NEXT
590 A$="  ALL STAGE CLEAR! SCORE"+STR$(SC)
600 B$="        TRY AGAIN(Y/N)?        "
610 LOCATE 0,3:PRINT A$;
620 FOR I=1 TO 15:GOSUB 650:NEXT
630 LOCATE 0,3:PRINT B$;
640 FOR I=1 TO 15:GOSUB 650:NEXT:GOTO 610
650 K$=INKEY$:IF K$="N" OR K$="n" THEN
CLS:END
660 IF K$="Y" OR K$="y" THEN 20
670 RETURN
680 '----- ｷｬﾗｸﾀｰ ﾃﾞｰﾀ -----
690 ' TIME BAR DATA
700 DATA 000000000000
710 DATA FE0000000000
720 DATA FEFE00000000
730 DATA FEFEFE000000
740 DATA FEFEFEFE0000
750 DATA FEFEFEFEFE00
760 DATA FEFEFEFEFEFE
770 ' STAGE 1 DATA
780 DATA 0002060E1E3F
790 DATA 3F7FFFFF6F27
800 DATA 331E0E060200
810 DATA 0000001CFFFF
820 DATA F9FFFFFFFFF9
830 DATA FFFF1C00000F
840 DATA 040C0C19B3EE
850 DATA BCB1BFBCB0BC
860 DATA FFB3296C44C4
870 ' STAGE 2 DATA
880 DATA F8F8F3E08824
890 DATA 200300FF380E
900 DATA 030000000000
910 DATA 0101051D3565
920 DATA C5FD03FD0505
930 DATA 85E53D0D0101
940 DATA 82C161323131
950 DATA 3032343234B2
960 DATA B0B23261C282
970 ' STAGE 3 DATA
980 DATA 030604060306
990 DATA 3E558888553E
1000 DATA 060306040603
1010 DATA C0203FD19010
1020 DATA 284582824528
1030 DATA 1090D13F20C0
1040 DATA 3C66C2E2267C
1050 DATA 90180F0F1890
1060 DATA 7C26E2C2663C
1070 ' STAGE 4 DATA
1080 DATA 000000000000
1090 DATA 0001FEFE0100
1100 DATA 000000000000
1110 DATA 1F0405087810
1120 DATA 20F00808F020
1130 DATA 10780805041F
1140 DATA E0404040E385
1150 DATA 897602027689
1160 DATA 85E3404040E0
1170 ' STAGE 5 DATA
1180 DATA 000000000101
1190 DATA C17E110C0301
1200 DATA 010100000000
1210 DATA 1F204F800F0F
1220 DATA 0F0FCD4820C8
1230 DATA 2D0F8F4F201F
1240 DATA B868381C541C
1250 DATA 949C9F951595
1260 DATA 959F343C68B8
```

★ベーマガ名作劇場にチョット期待。僕は比較的新しいベーマガ読者なので，過去の投稿作品をいろいろ見てみたいのです。（神奈川県川崎市・**色落ち18歳**）……**【編：**実際に体験できるといいんだけどね。**Dr.D：**名作劇場で紹介した作品の移植投稿も待っておるぞ。**】**

マイコンBASICマガジン

明日のスター・プログラマー

Programmer's Room

今月プログラム賞

毎月すべての掲載プログラムの中から，アイデア，リストの面で優れている作品を，それぞれ「ベストアイデア」「ベストリスト」，そして最も優れていると思われる作品を「ベストプログラム」として選考していきます。

それでは，今月の受賞作品を，編集部からのコメントとともに振り返ってみましょう。

ベストプログラム	C++Builder用	Flied Egg	赤羊旅団(17歳)
コメント：細かな部分で気になる点が多いものの，ユニークなゲームデザインと，卵が割れるなどの処理が評価されてのベストプログラム受賞です。			
ベストアイデア	Visual Basic用	ジョーピン	大石　晋也(21歳)
コメント：身近な遊びをゲーム化した点が，編集部内でも非常に好評でした。そのゲームアイデアの着眼点が評価されてのベストアイデア受賞です。			
ベストリスト	HSP用	レースdeぽん	寺内　洋平(31歳)
コメント：約200行，容量では4Kバイトを切る短いリストで，ゲームプログラムを制作した点が評価されてのベストリスト受賞です。			

投稿ありがとう

いしかわね　みひろ／大平瑞穂／ARQEE／根本賢樹／ZAP／まりもん／エレフ／金井義男／橘　一馬／中島武志／古川和広／山口晴久／根本　浩二郎／田村なおき／若林吾郎／Danny／佐橋晴久／中丸博之／笹目浩雄／田中士郎／光田賢治／でむぱ／池野　健一郎／笹木／佐藤雅人／緒方幸雄／クォータニオン／森田健一／張間秀幸／伊藤　弘／菱刈優／DayDream／若林　猛／寺田　昇太郎／増田好春／Campus／横山正隆／高田俊次／松本幸雄／佐野　真一郎／飯合文博／久保田　利勝／村井貴之／佐野文博／ソケット／田辺一之／笹原康夫／飯島和馬／永富和久／田辺孝次／林　雄一／直井一郎／阿部和也／平松優／本田修哉／緒方　俊／三木祐二／伊藤　大／新田達也／武藤雄哉／
※紹介できなかった方ごめんなさい

読者からのご意見

編：ここでは，投稿応募カードの裏面や投稿説明文などで，今までいただいた投稿者からのご意見を掲載していきます。

ゆーとんパパさん（33歳）

今月号掲載作品『パズルナイン』投稿者

私とベーマガとの最初の出逢いは，中学の頃だったように記憶しています。

当時はプログラムの意味も解らず，ベーマガを見てプログラムを打ち込んで，ゲームを楽しんでいました。

そんな私が，５年ほど前にWindowsのPCを購入し，ベーマガと再会しました。なんかすっごく懐かしく嬉しくなりました。その後，すぐにVisualBasicを購入し，ベーマガを購読しながらの独学をはじめました。

今回の作品は，ポケコン用プログラムでの投稿ですが，やっと投稿することができました。次回はぜひとも，Visual Basicでチャレンジしたいと思います。

編集部より

インターネット上の掲示板などでも，昔ベーマガを見ながらプログラムを打ち込んで，ゲームを遊んでいた，という話しを見かけたりします。

そういった昔からの読者が，現在も投稿で参加してくださっているというのは，ホントにありがたいものです。

あなたの回りでも，昔ベーマガを読んでいたという人はいませんか？もし，昔の読者さんがいたら，ぜひ現在のベーマガを紹介してあげてくださいね。

あと，ゆーとんパパさん。VBプログラムの投稿作品も，お待ちしていますよ。

ベーマガ名作劇場

このコーナーでは，今まで投稿プログラムコーナーに掲載された作品の中から，編集部が名作として選んだ作品を紹介していきます。

今月の名作：『GOKIVADER』(1988年11月号掲載　末冨 悟 作 他にも移植版などあり)

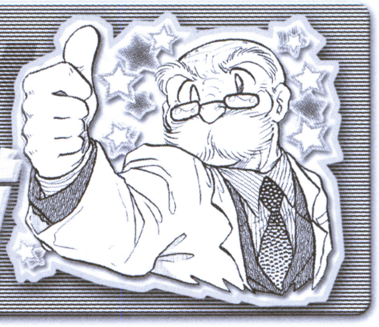
●ノート

一見インベーダー風のシンプルなゲームだが，“ゴキブリ”を題材にしている点が受けたのか，はたまた遊びやすいゲーム内容とシンプルなリストが評価されたのか，今までに移植版・改造版の投稿作品がコンスタントに掲載されてきた。

掲載時のCHECKER FLAGで，Dr.Dが「シンプルでリストも短い!!わしが求めていたのはこんなゲームじゃ!!」というコメントをしている点からも，このゲームの内容の良さが伺える。

ちなみに今月号までにおける『GOKIVADER』最新版は，1999年４月号掲載のPC-88用バージョンである。現行機種での移植版投稿が待たれるところだ。

Dr.Dのベスト・プログラムへの道



今月のテーマ：『リストにコメントを記述しよう』

今月は，「リストにコメントを記述しよう」をテーマに，コメント記述の必要性やコメント記述に関するテクニックなどを解説していこう。

コメント記述をしっかりマスターして，一歩進んだプログラマーを目指してほしい。

(Dr.D)

コメントの必要性を理解しよう

コメントが必要な一番の理由は，プログラムの命令や処理の内容をわかりやすくするため，と言えるじゃろう。

コーディングを行っているときには，処理を理解しているため，面倒でコメントを記述しなかったが，他の作業を繰り返している間に，記述した時には理解していた処理がどういった内容だったか忘れてしまった，という経験がある読者も多いのではないかな？

こういった事を繰り返していては，プログラミングの作業がたびたび中断して，作業効率が下がってしまう。

適切なコメント記述をすれば，プログラミングの作業効率も上がるし，他の人が見ても理解しやすく，またプログラム・リストをより見やすいものにできるのじゃ。

コメントを記述してみよう

コメントの必要性がわかったところで，**リスト1**を使用して，実際にコメントを記述する過程を紹介していこう。

まず，注目してほしいのが，このプログラムは何をするのか，ということ。このリスト内には簡単な処理しかないが，それでもすぐには理解できないじゃろう。

こういった事もコメント記述がないのが原因。まずは，リストの最初にプログラムの処理内容を記述してみよう(**リスト2**)。

次に，基本的なコメントとして，プログラム内で使用している変数や定数，命令や行っている処理内容に対する注釈を入れたものが，**リスト3**になる。

工夫したコメント記述を行おう

基本的なコメントは，**リスト3**のような記述で充分なのじゃが，ここからもう一歩踏み込んで，そのコメントの記述方法にも工夫を凝らしてみたいと思う。

今までに，**リスト4**のようなコメント記述を見たことがあるじゃろうか？

中には，「なにカッコつけて面倒な記述しているんだよ！」なんて思う読者もいるかもしれんが，それは大きな間違い。この一見カッコつけた記述こそ，工夫した記述と言えるのじゃ。

この記述方法は，主に「区切り」として使用される。つまり，このような目立つコメントにすることで，より目に付きやすくしておるのじゃ。

この「区切り」記述は，主に関数の頭や，まとめて変数を定義する部分などで使用される。この記述を適用したものが**リスト5**になる。

もう一つ，工夫したコメント記述として行ってほしいのが，制作情報じゃ。これは，「いつ・誰が・どんなプログラムを制作したのか」といったことをリストの最初に記述することで，リストそのものに対して基本的な情報をもたせることができる。この制作情報を記述したのが**リスト6**になる。

さて，ここまで紹介してきたコメント記述を行うことで，まったくコメントのない**リスト1**が，**リスト6**のように格段に見やすいリストに仕上がったのが，実感できるのではないかな？。

このコメント記述を面倒がらずに常に行うクセをつけ，その上でより見やすくなる記述を行うように心がければ，一歩進んだプログラマーであるといえるじゃろう。

掲載プログラムを改造してみよう

このコーナーでは，毎月掲載されたプログラムの中から1～2本ピックアップして，そのプログラムの改造を紹介していきます。

HSP用『レースdeぽん』を改造しよう

●改造その1「自車を無敵状態にする」

これは，プログラム改造を行う上で，定番と言えるものじゃ。

改造方法は，157行目にある「BOM=1:BOMB=1:BN=I:SP=SSP」という行をコメントアウト(最初に;を付ける)すればOKじゃ。

●改造その2「縁石で走行可能にする」

ゲーム中，自車はコース内のみ走行可能で，縁石に乗り上げて走ることができない。そこで走行範囲を広げて，縁石でも走行可能にする改造(**写真1**)を紹介しよう。

改造は，**リスト7**のように記述すればOK。要は，移動の制限範囲を書き換えるというわけじゃ。

また，このゲームのスタート時の処理を流用して，縁石走行中は，得点が入らないようにもしてある。

《リスト1》コメント記述のないリスト

```
Private Sub Command1_Click()

    Const F = 1
    Const Max = 5

    Dim a As Integer
    Dim i As Integer

    Text1 = ""

    If F = 1 Then
        a = 0
        For i = 1 To Max
            a = a + i
        Next i

    ElseIf F = 0 Then
        a = 1
        For i = 1 To Max
            a = a * i
        Next i

    End If

    Text1 = Str(a)

End Sub
```

《リスト2》処理内容コメントを記述したリスト

```
'1～Maxまでの足し算，もしくは掛け算を求める
Private Sub Command1_Click()

    Const F = 1
    Const Max = 5

    '(以下省略)
```

《リスト3》基本的なコメント記述を行ったリスト

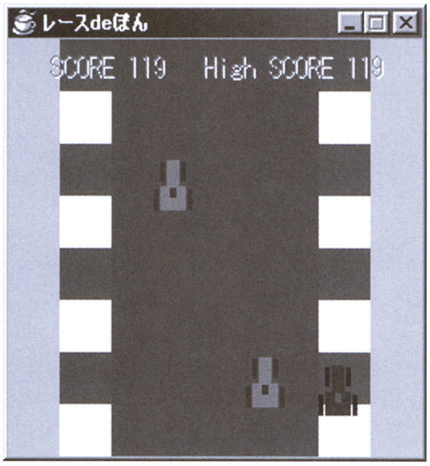
```
'1～Maxまでの足し算，もしくは掛け算を求める
Private Sub Command1_Click()

    '定数を定義
    Const F = 1    'F=1なら足し算，F=0なら掛け算を行う
    Const Max = 5  '計算を行う最大の数

    '変数を定義
    Dim a As Integer  '計算の答え
    Dim i As Integer  'For文用

    'テキストボックス内を初期化
    Text1 = ""

    '1～Maxまでの足し算を行う
    If F = 1 Then
        a = 0
        For i = 1 To Max
            a = a + i
        Next i

    '1～Maxまでの掛け算を行う
    ElseIf F = 0 Then
        a = 1
        For i = 1 To Max
            a = a * i
        Next i

    End If

    '答えをテキストボックスに表示
    Text1 = Str(a)

End Sub
```

《リスト4》区切りコメントの記述例

```
'----------- サンプルプログラム1 ------------

'********************************************
'*          サンプルプログラム2            *
'********************************************
```

《リスト5》区切りコメントを記述したリスト

```
'--------------------------------------------
' 1～Maxまでの足し算，もしくは掛け算を求める
'--------------------------------------------
Private Sub Command1_Click()

    ' ------------ 定数を定義 ------------
    Const F = 1    'F=1なら足し算，F=0なら掛け算を行う
    Const Max = 5  '計算を行う最大の数

    ' ------------ 変数を定義 ------------
    Dim a As Integer  '計算の答え
    Dim i As Integer  'For文用

    '(以下省略)
```



▲《写真1》HSP用『レースdeぽん』改造2を行った画面

《リスト6》制作情報を記述したリスト

```
'*****************************************
'* 簡単な計算プログラム(Visual Basic用)  *
'* 制作者：Dr.D  最終更新日：2002 11/24  *
'*****************************************

'-----------------------------------------
' 1～Maxまでの足し算，もしくは掛け算を求める
'-----------------------------------------
Private Sub Command1_Click()

    ' ------------ 定数を定義 ------------
    Const F = 1    'F=1なら足し算，F=0なら掛け算を行う
    Const Max = 5  '計算を行う最大の数

    ' ------------ 変数を定義 ------------
    Dim a As Integer  '計算の答え
    Dim i As Integer  'For文用

    'テキストボックス内を初期化
    Text1 = ""

    '1～Maxまでの足し算を行う
    If F = 1 Then
        a = 0
        For i = 1 To Max
            a = a + i
        Next i

    '1～Maxまでの掛け算を行う
    ElseIf F = 0 Then
        a = 1
        For i = 1 To Max
            a = a * i
        Next i

    End If

    '答えをテキストボックスに表示
    Text1 = Str(a)

End Sub
```

《リスト7》HSP用『レースdeぽん』改造その2のリスト

```
*car
;★改造部分
;元々(X>61)と(X<162)だった部分を以下のように
;書き換えて縁石部分まで移動可能にする
if (KY&1) :if (X>30)  : X=X-1
if (KY&4) :if (X<195) : X=X+1

;(途中省略)

*check
if IC!1      :return
if Y.i<172   :return
if Y.i>230   :return

;★改造部分
;得点計算を行うif文に,「if X<180」という条件を加えて,
;縁石走行中に得点を加算させないようにする
if X>59 : if X<180 : if SP=5 : S=S+1
if X>59 : if X<180 : if SP>1 : S=S+1 : if HS<S : HS=S
```

プログラムに関する Q&A コーナー

プログラミング公開質問状

このコーナーでは，プログラミングに関する読者からの質問にお答えします。プログラムを組んでいてうまくいかないことや疑問点などがありましたら，どしどし質問をお寄せください。応募方法は記事巻末を！

Q VB6.0で，イライラ棒風ゲームを作っています。LineメソッドでFormに線を描いて，マウスカーソルの現在位置の色で，線の上からはみだしたかどうか判定しようと思っているんですが，方法が分かりません。教えていただけないでしょうか？

もし，そのような方法がとれない場合，違う良い方法を教えてくださるとうれしいです。

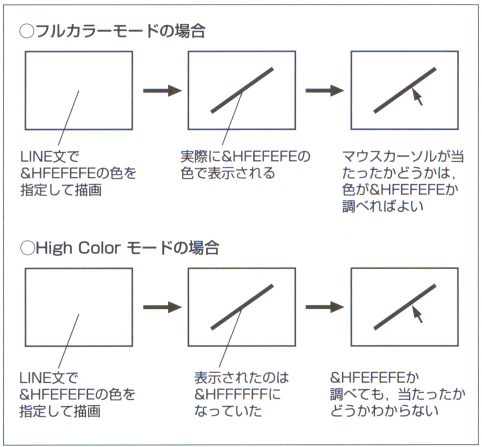
【神奈川県小田原市・漆畑龍典14歳】

A VB6.0では，PointメソッドでPictureBoxやFormの指定した点の色を知ることができるので，これを使うのがいいでしょう。

また，現在のマウスカーソルの座標を知るには，FormのMouseMoveイベントで返されるXとYの値を利用するという方法があります（**リスト1**）。

《リスト1》マウスカーソルが示す色を取得するサンプル・リスト

```
Private Sub Form_MouseMove(Button As
Integer, Shift As Integer, X As Single, Y
As Single)
  C=Point(X, Y)

  ～色に応じた処理～

End Sub
```



《第1図》フルカラーモード以外では，指定した色で表示されない可能性がある

注意点として，Windowsの画面モードがフルカラー（24bit以上）以外の色数（256色や6万色など）の場合，Lineメソッドで描画する際に指定した色と，Pointメソッドで読み出した値が必ずしも一致するとは限らないことです（**第1図**）。

たとえば，Windowsの画面モードがHigh Color（16bit）などに設定されている場合（**写真1**），**リスト2**のプログラムを実行すると，指定色と異なる色で描画されることがわかります（**写真2**）。

《写真1》Windowsの画面モードをHigh Color（16bit）にしてみる

《リスト2》指定した色で描画されるかどうかのテスト・プログラム

```
Sub Command1_Click ()
  c1 = &HFEFEFE
  PSet (0, 0), c1
  c2 = Point(0, 0)
  MsgBox Hex(c1) + " : " + Hex(c2)
End Sub
```

これに対処するには，フルカラーモードならば必ず指定した色で描画されるので，フルカラー専用のプログラムとしてしまうという方法がありますが，これは動作環境を限定してしまうのであまりよくありません。

他の方法としては，色が完全に一致せずとも，ある程度の範囲で判断できるようにするといったものもあります。

これは，**リスト3**のように，色を判別する際に完全にイコールでなくとも，ある程度の範囲内におさまっていればそれを受け入れるというもので，判断するのと近い色を使わないようにすれば，たい

《写真2》HighColorでのリスト2の実行結果。指定した色と，実際に描かれた色が異なることがわかる

《リスト3》変数cの色情報を，RGBの各成分に分割して許容範囲かどうか判断するサンプル・プログラム

```
c=Point(x,y)

'色を完全一致で判断する例
If c=&HC1FE67 Then ～

'ある程度許容する例
r=c And &HFF
g=(c ¥ 256) And &HFF
b=c ¥ 65536
If b>&HBB And b<&HC6 And g>&HF9 And
r>&H62 And r<&H6C Then ～
```

てい問題は解決すると思います。このリストでは，変数cに入っている色情報を，RGBの各成分に分割して，それぞれが許容範囲かどうか判断しています。

【回答者：やまかんこと山崎寛】

Q Delphiでシューティングを作っています。ところで，オブジェクト指向風に書くのがよいのですか？　それとも，ダーッとBASICライクに書いたほうがよいのですか？　長所，短所が知りたい。

【兵庫県朝来郡・稲津和磨17歳】

A これは，2003年1月号のOFで掲載された質問です。

オブジェクト指向というのは，構造化プログラミングをより進めた考え方で，この考え方に沿ってプログラムを設計すると，自然とプログラムは特定の処理を行う細かいパーツの集合のような形になっていきます。

しっかりと考えて細分化されたプログラムというのは，末端のコードを追いかけなくても全体の流れをつかみやすくて，読みやすい，わかりやすいプログラムになります。読みやすいプログラムというのは，修正や改造が楽になるので，自分のためにも他人のためにも重要です。

また，プログラムをパーツ単位で細かくしていくことで，個々のパーツの独立性が高くなり，後日そのパーツを使いまわして楽にプログラムできるようになったり，あるいは多人数での開発を行う場合などに分業がしやすいといったこともあります。

短所はあまり思いつかないのですが，プログラムが冗長になりがちなことや，ある程度しっかり設計が必要になるので，いきあたりばったりでプログラミングできないことでしょうか。それほど短所と思うようなことではないかもしれません。

昔のBASICプログラムの多くが構造的でなかったのは，言語の仕様としてローカル変数や複数のソースを持てなかったことや，動作速度やメモリの制限の問題から，わかりやすさよりも速度を優先させていたためです。プログラムをわかりやすく書く＝プログラムが遅くて長くなる，という風潮もあったと思います。

現在では，わかりやすいプログラムにしたために，その差がわかるほど速度に影響が出ることはほとんどありませんし，多少リストが長くなっても問題が生じることはないでしょう。可能ならば，後日の自分のためにも，また作ったプログラムを見る誰かのためにも，オブジェクト指向の考え方に基づいてわかりやすいプログラムを心がけたほうがいいと思います。

とくに，プログラムの規模が大きくなるほど影響は大きいですよ。

【回答者：やまかんこと山崎寛】

あなたの疑問・質問を受け付けます

質問事項／お名前／年齢／職業（または学年）／郵便番号／住所／連絡先（電話番号，メールアドレス）を記入し，以下の宛先までお送りください。

■ハガキ・封書での応募

宛先：〒141-8715東京都品川区東五反田1-11-15　電波新聞社マイコンBASICマガジン編集部「プログラミング公開質問状」係

■メールでの応募

webmas@basicmagazine.net　（件名を「プログラミング公開質問状」係としてください）

■応募する際の注意点

①質問はわかりやすく具体的に書こう

質問の中には，「内容が抽象的すぎてどう答えたらいいかわからない」といった内容のものも見受けられます。

たとえば「シューティングゲームの作りかたを教えてください」と質問されても，どんなシューティングゲームの，どんな部分の作りかたについて答えればいいのか判断がつきません。

これが「現在，シューティングゲームを作ろうと考えています。キー入力に応じてキャラクターを動かすことはできたのですが，弾を発射するやりかたがわかりません。キーを押すと自分のキャラクターの場所から前方に弾が飛んでいくようにするにはどうすればいいのでしょう？　また，連射できるようにする方法も教えてください」なんて書きかたならば，だいぶ具体的に質問の内容がわかり，答えやすくなります。

このようにまずは，何がわからなくてどう悩んでいるのか，どう答えてほしいのかを整理してみましょう。

また，どんなパソコン，どんなプログラミング環境なのか，どれくらいプログラミングの経験があるのかなども回答に必要になる場合があります。できれば自分の使っている環境（パソコンやプログラミング言語の種類など）も一緒に書くようにしてください。

②作りたい作品のイメージを伝えよう

おそらく，誰でも自分の作りたいゲームなりアプリケーションなどの完成形を「こんなものを作りたい」とイメージしていると思いますが，これもできる限り正確に伝えることが必要になります。

ゲームのジャンルを指定されても，たとえばRPGと一口にしても『ドラゴンクエスト』や『ウィザードリィ』といったいろいろなタイプのゲームがありますから，そういった中で質問された方がどんなゲームを想定しているのかこちらにはわかりません。

図などを交えて説明したり，既存のソフトに近い内容ならば，そのソフト名などを挙げて，たとえば「『グラディウス』のオプションのように後ろについてくる動きを作りたい」などと質問してもらえると，イメージがより伝わりやすくなります。

■連載■ PCハードウエアテクニック

PC製作にチャレンジするための知識＆テクニックを徹底解説

誌上パソコン組み立て教室

九十九電機の初心者用セット

パーツを一つずつ厳選してカスタムマシンを組み立てるのも面白いのですが，ビギナーや手っ取り早くできるだけ安価に…ということであれば，パーツキットやセットを利用したDOS/V機の自作がオススメです。CPUからメモリ，さらにケースまで製作に必要なパーツがすべてセットになったDOS/Vキットなら，パーツの買い忘れの心配もなく，誰でもPC製作が楽しめます。今回は，市販PCに引けを取らない高性能マシンをDOS/V自作セットで製作してみました。

1 お店が選んだセットで高性能DOS/V機を作ろう!!

手軽に高性能マシンが自作できるパーツキット

DOS/Vパーツキットは，PCショップで販売されていますが，今回は作りやすさの点や完成後の性能，さらに充実したサポートサービスから，九十九電機に初心者用としてパーツをそろえてもらいました。

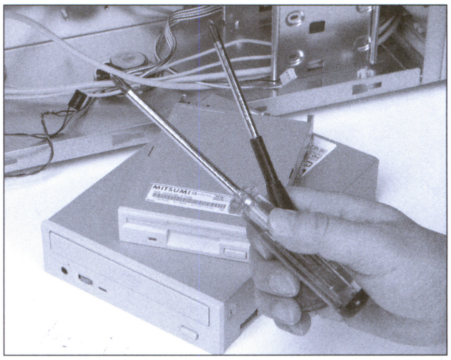
スペックやパーツ内容については，写真や説明のとおりですが，とくに注目したいのは，インテル社の「845GE」チップセット採用のマザーボードです。

この845GEは，グラフィック機能を持ち，高速メモリのDDR SRRAMやUSB2.0をサポートする高性能＆多機能な統合タイプで，最新マザーボードに数多く採用されている人気のチップセットです。

さらに，マザーボードのオンボード機能として，サウンド入出力，LANインターフェースを備えています。そのため，LANカードやサウンドカードは不要で，その分，組み立て作業が簡単というメリットもあります。

パーツ構成は，115ページのとおりですが，マザーボードのCPUソケットは，Socket 478タイプで，Celeron（セレロン）1.7GHzをPentium（ペンティアム）4に交換することも簡単で，完成後のアップグレードの楽しみもあります。

ベテランのDOS/V自作派には，本機のようなさまざまな機能を織り込んだオールインワン・タイプのマザーボード・マシンよりも，パーツを個別に揃え製作するカスタムマシンに人気があるよ



▲DOS/Vパソコンの自作といっても作業はいたって簡単。工具類はプラスドライバー（大小2本があればベスト）だけで大丈夫

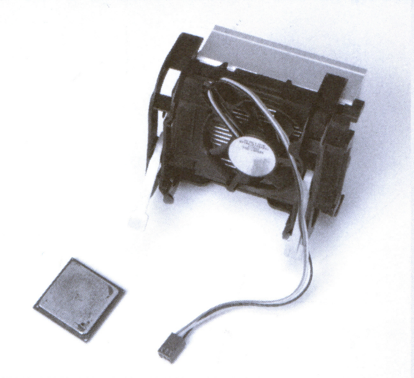
九十九電機の一押しDOS/V機自作セット

今回紹介したパーツは，DOS/Vショップの老舗，九十九電機にそろえてもらった初心者用セット。作りやすさと優れたコストパフォーマンスが特徴。Celeron 1.7GHzと最新845GEチップセット採用のマザーボード，DVD-ROMドライブなど，低価格なモデルながら，充実したパーツ構成は魅力といえる。今年1月12日に開催のマイコンBASICマガジン主催「自作パソコン組み立て教室」でもこのセットを基本に採用されている。

主なスペック
CPU：インテルCeleron1.7GHz，メモリ：128MB DDR-SDRAM，HDD：40GB，DVD-ROMドライブ：最大52倍速DVD/CD-ROM，LAN：10BASE-T／100BASE-TX，サウンド機能：AC97準拠，OS：Windows XP HE，キーボード・マウス・スピーカ付属

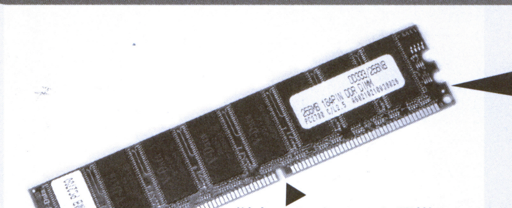
DOS/V マシン製作の主要パーツ（仕様は変更になる場合もあります）

CPU（中央処理ユニット）



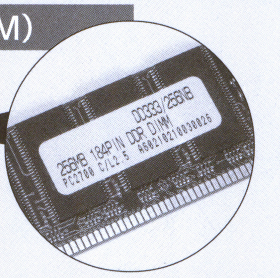
▲マシンの処理性能を決めるCPUには，Celeron1.7GHzを採用。一般使用では，まったく問題ないポテンシャルだが，Pentium4へのアップグレードも可能

拡張メモリモジュール（DDR SDRAM）



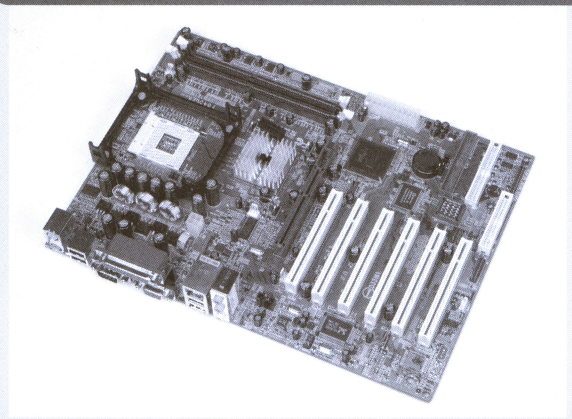
▶従来のSDRAMと同様のソケット形状だが，DDR SDRAMでは切り欠き部分が一つ

▲グラフィック系ソフトなどで大容量データを扱う場合は，高速な転送速度を持つDDR-SDRAMが最適



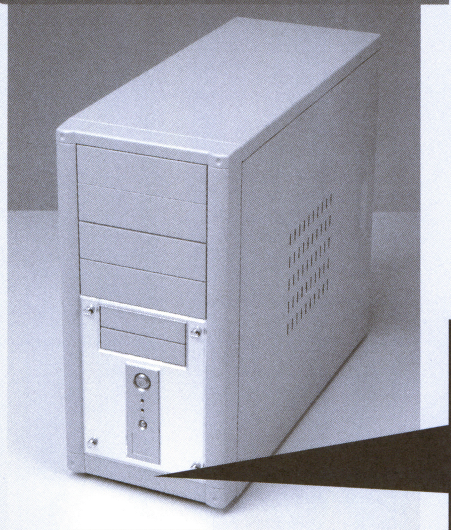
▲DDRとはダブルデータレートの略で，これまでのSDRAMのほぼ2倍の転送性能がある。自作セットでは128MBタイプ

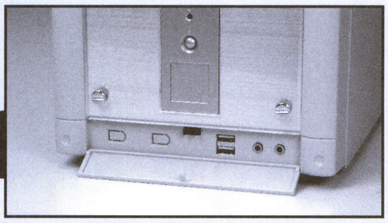
マザーボード（845GEチップセット）



◀インテル社の最新統合チップセット「845GE」採用のマザーボード。DDR SDRAM対応や簡単なケーブル接続だけで高速なデータ転送が可能なUSB2.0インターフェースなど，多彩な機能が特徴。PCIスロットは6基あり，拡張カードによるアップグレードにも余裕を持って対応できる

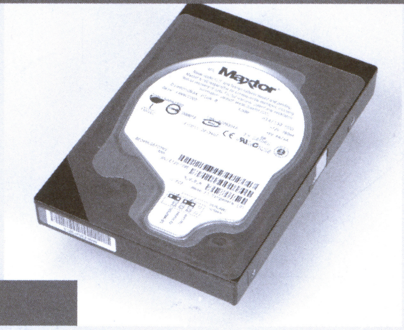
PCケース（ミニタワー型）





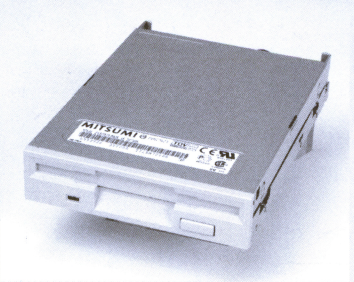
▲使い勝手に直結するのがPCケースのデザインや仕様。フロントパネルからオーディオ入出力やUSBケーブルが接続可能で，シンプルながら高級感のある飽きのこないスタイルには好感が持てる

ハードディスク・ドライブ（HDD）



▲高画質なグラフィック・データや動画データなどの大容量ファイルでも安心して保存できる40MBタイプを採用。接続規格はIDE

フロッピーディスク・ドライブ（FDD）



◀データの受け渡しなどには，FDDが欠かせない。720kBと1.44MBフォーマット対応の2モードタイプを用意

DVD-ROMドライブ

◀DVDビデオの再生が楽しめるDVD&CD-ROMドライブ。OSやアプリケーションのインストールにも欠かせないパーツだ

うですが，一般の使用目的ならオールインワン・タイプでも，性能や機能面で不足を感じることはあまりなく，コストパフォーマンスでは，ダントツに優れています。

とくにこれから自作DOS/V機にチャレンジする人には，オールインワン・タイプのDOS/Vキットは，ベストな選択といえるでしょう。

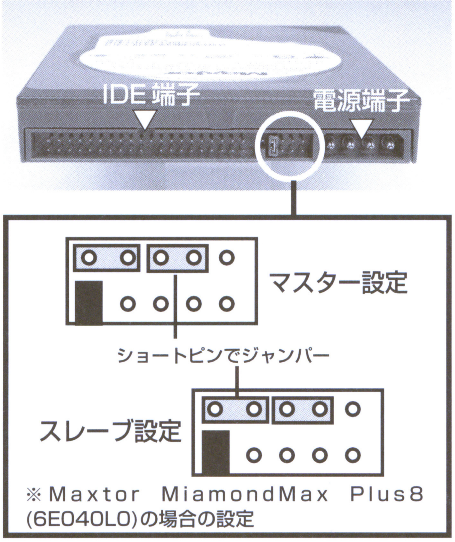
2 ドライブ類の下準備

まず，PCケースに組み込むハードディスクやCD（DVD）-ROMなどのドライブの下準備から始めます。これらの内蔵ドライブは，マザーボードとIDEインターフェースでデータのやりとりを行うための「マスター」「スレーブ」設定を行うことになります。この設定は，IDE特有のドライブ識別方式で，ケーブル接続をするIDE端子の隣りに設けられたジャンパーピンで「マスター」「スレーブ」のいずれかを選択し，IDEケーブルごとに設定がだぶらないようにします。マザーボードには，通常「プライマリ」と「セカンダリ」の2基のIDE端子があり，最大4台のIDE機器が接続可能です。転送性能では，右図のようにHDDはプライマリ側，CD-ROMはセカンダリ側と端子を使い分けた方が有利です。

よくあるトラブルと対策法

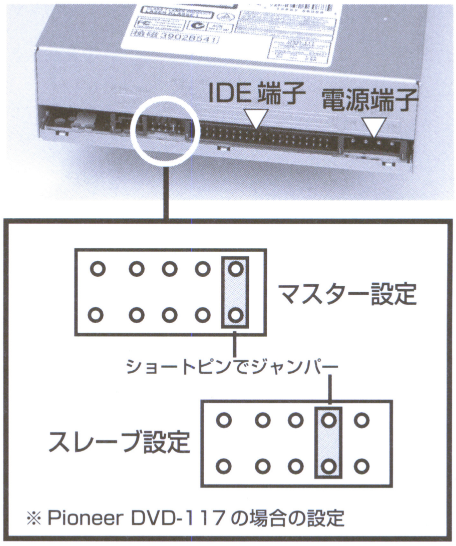
『完成後にHDDやCD-ROMドライブを認識しない』
●対策：ケーブル接続が完全であれば，ドライブの「マスター」「スレーブ」設定が正しいか確認する

HDDのIDE設定



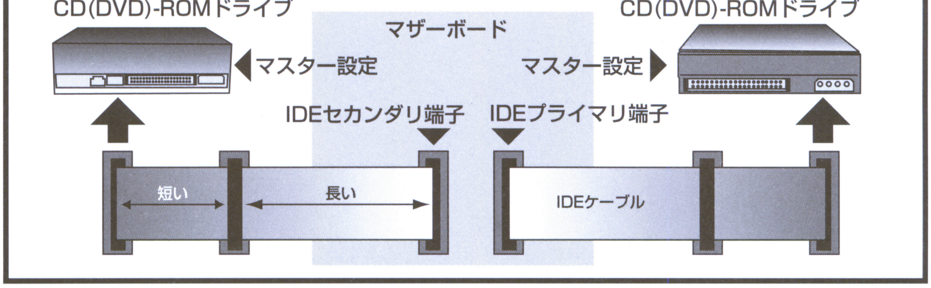
※Maxtor MiamondMax Plus8 (6E040L0)の場合の設定

CD-ROMドライブのIDE設定



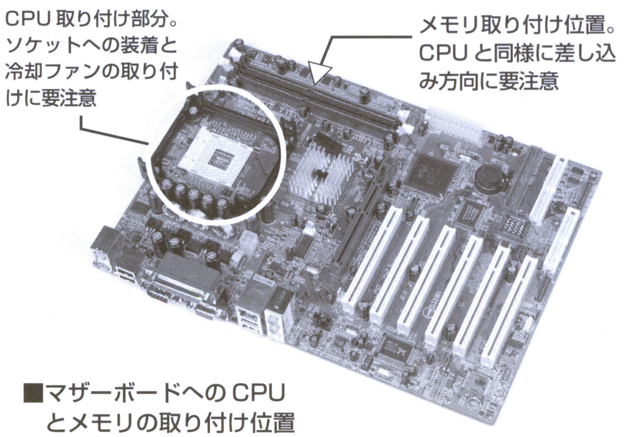
※Pioneer DVD-117の場合の設定

■マスターとスレーブ設定ドライブの接続例



3 マザーボードへのパーツの組み込み

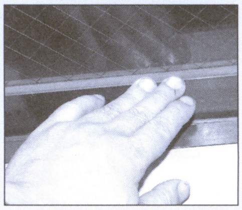
パーツの組み込みは，マザーボードへのCPUとメモリの取り付けから始めましょう。PCケースへマザーボードを取り付けた後から行う人も多いようですが，スペースが狭く，取り付け後の確認もしづらいので，オススメできません。CPUとメモリは，端子がむき出しで最悪の場合，静電気などで突発的に壊れることもあります。そのため，取り付け作業は，指で端子に触らないなど細心の注意を払って行うようにしましょう。また，差し込み方向がありますので，ソケットと端子の形状を確認して差し込んでください。入りにくいからといって無理矢理差し込むことは絶対に避けてください。CPUでは端子ピンが折れると，動作しないどころか二度と使えなくなります。



■マザーボードへのCPUとメモリの取り付け位置

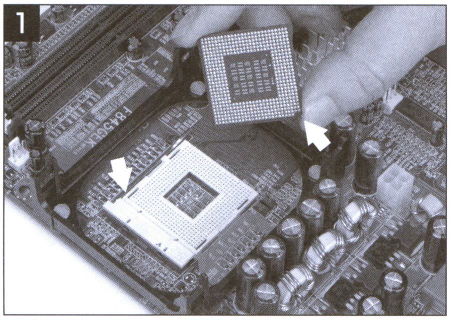
Socket 478タイプのCeleronは，極めてコンパクトなCPUですが，1.7GHzの高速な動作を行うために発熱量が多く，CPU形状に似つかわしくない大型の冷却ファンが付属しています。しかし，取り付け方法を間違えると，効果的な放熱

身体の静電気を放出しておこう

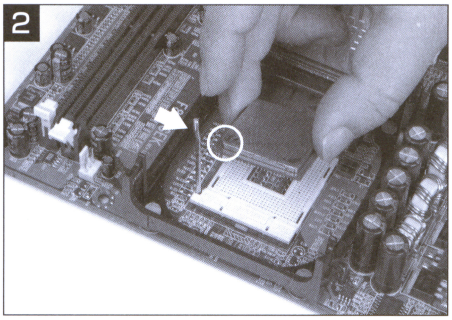


冬のように空気が乾燥する時期は，静電気がDOS/V機製作の大敵。身体に帯電した静電気は，アルミサッシや机などの身近な金属部分に指で触れることで放出することができる。組み立て作業に入る前に，念のために行っておこう。

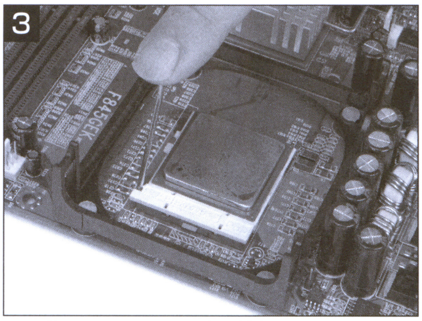
CPUの取り付けと冷却ファンの設置



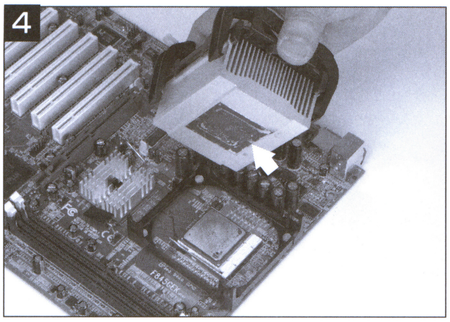
▲ソケットとCPUの端子形状を確認。ピンが2本抜けている部分（矢印）で差し込み方向を調べる



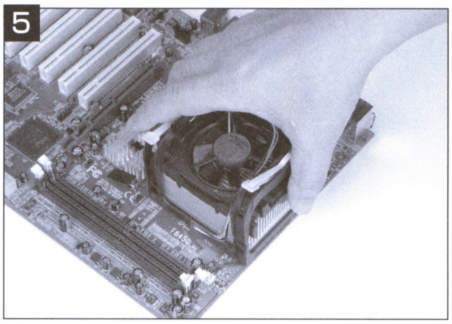
▲ソケットのロックレバー（矢印）を垂直方向に起こし，CPUを慎重に差し込む。CPUにある切り欠き（丸）は，ピンのない部分



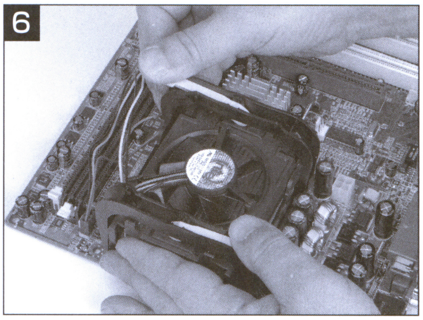
▲ソケットとCPUに隙間がなく密着する位置まで差し込んであることを確認し，ロックレバーを倒し，CPUを固定



▲大型のCPU冷却ファンを装着する。ファンのCPUとの密着部分に塗られている熱伝導材（矢印）は，間違って拭き取らないこと



▲均等に上から少しずつ力を加えて，ソケットの周りを囲む冷却ファン固定ガイドにカチッという手応えがあるまで押し込む



▲冷却ファン上部の左右にあるロックレバーを同時に水平位置まで倒して，ファンを固定

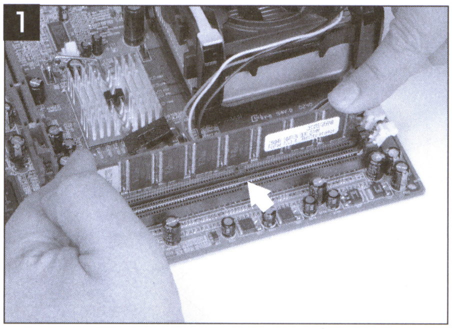
ができなくなり，CPUが動作しなくなることも考えられます。冷却ファンに限らず，「パーツの取り付けを慌てず正確に行う」ことが，DOS/V機製作のコツといえます。

よくあるトラブルと対策法

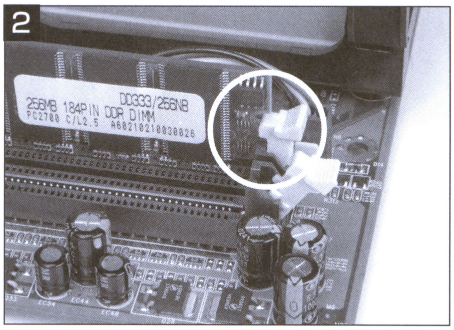
『電源は入るがPCディスプレイが真っ暗なままで何も表示されない』『内蔵スピーカから警告音が鳴り起動しない』

●対策：CPUとメモリの取り付けを確認。なかでもメモリの接触不良による場合が多く，ソケットから抜き，再挿入することで解決することもある

メモリの取り付け



▲差し込み方向を端子の切り欠き位置（矢印）で確認し，マザーボードのメモリソケットに左右均等に力を加えて押し込む



▲正しい位置までメモリが差し込んであれば，ソケットの左右に設けられたレバーが自動的にメモリ基板をロック（丸の状態）

4 PCケースへのパーツの取り付け

DOS/V機製作で一番手間がかかる作業が，PCケースへの組み込みです。とはいっても，手際よく行えば短時間で済ますこともできます。狭いスペースしかないPCケース内部の作業になるため，大きなパーツから順番に取り付けていく「手際の良さ」を考慮しないと，前に設置したパーツが邪魔で，取り付けができないこともあります。最もスペースをとるマザーボードから開始し，CD-ROMドライブ，FDDとHDDの順にすすめていけばよいわけです。

これらの作業のほとんどが，パーツのネジ止めですが，DOS/V機の自作にはとても重要なことなのです。ドライブ類は高速回転するモータを搭載しているわ

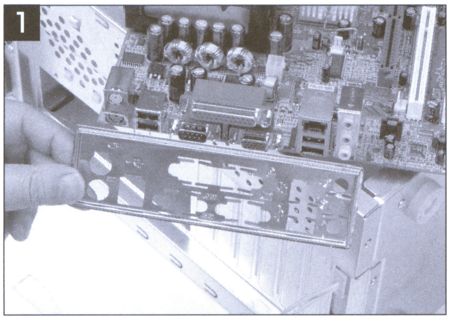
よくあるトラブルと対策法

『内蔵ドライブの周辺からの耳障りな動作音や異常音が聞こえる』

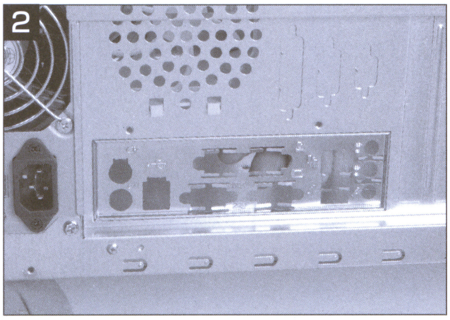
●対策：HDD，FDD，CD-ROMの各ドライブを固定しているネジの締め直しを行う。力まかせに締めるとネジが壊れてしまうこともあり，「適度な力で」が鉄則

けですが，最近のドライブはモータの振動は軽減されているとはいえ，固定ネジの締め具合の良し悪しで，トラブルの原因になることも少なくないのです。とくに無視できないのが，振動による耳障りな動作音で，自宅などの静かな環境で使う場合はとても気になるものです。必要以上の締めすぎは禁物ですが，確実に固定できているか確認しておきましょう。

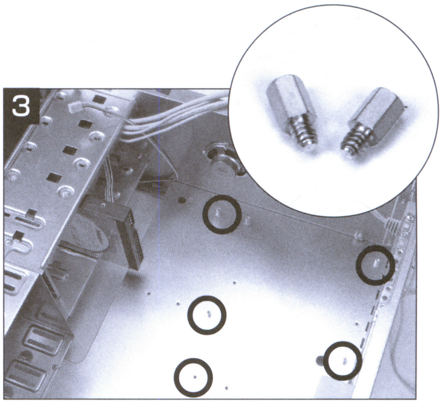
マザーボードの取り付け



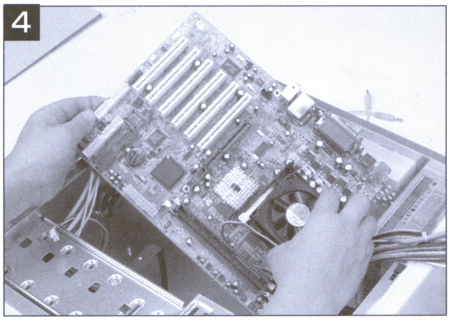
▲自作キットに付属の端子パネルを用意し，マザーボードの端子位置を確認



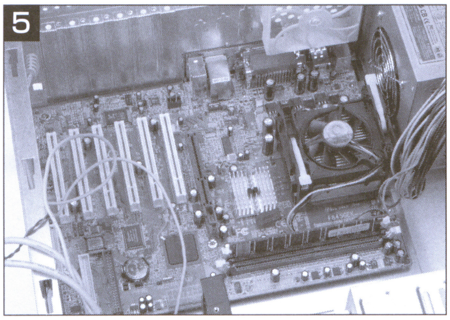
▲端子パネルをPCケースのリアパネルに取り付ける。ケース内側からはめ込む



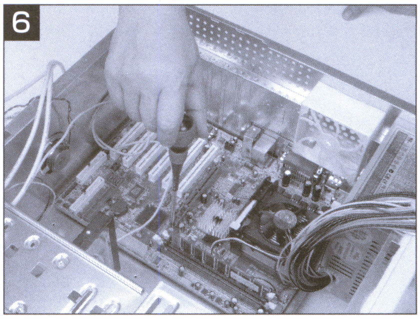
▲マザーボードのネジ穴に合う PC ケースのネジ位置に専用の固定ネジを設置



▲電源ケーブルを外に引き出して，マザーボードを PC ケースに収める



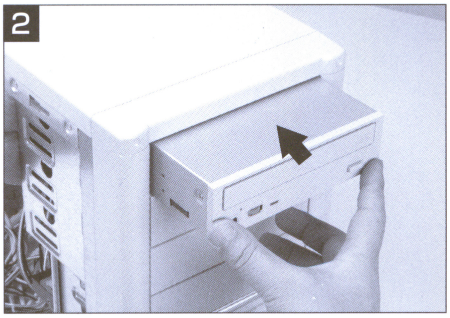
▲リア端子や PCI スロットの位置を合わせながら所定の位置にマザーボードを配置



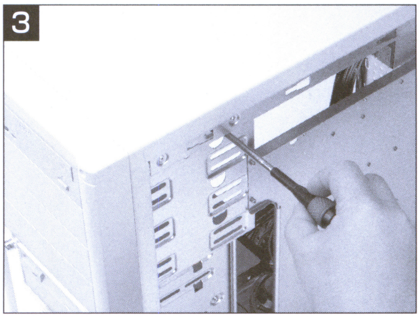
▲マザーボードを PC ケースの各ネジ穴に合わせて固定していく

ドライブ類の設置

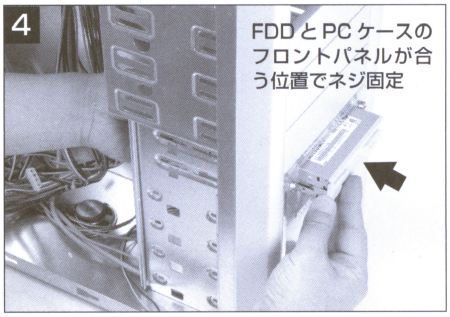
▲CD(DVD)-ROM ドライブ用の５インチ・ドライブベイのフタを外す



▲CD(DVD)-ROM ドライブを少しずつ力を加えながら５インチベイに押し込む

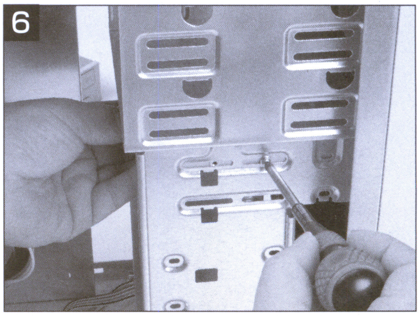


▲５インチベイの両側から４本のネジで確実に固定。緩いと動作音が大きくなる



▲3.5 インチベイのフタを外し，前後の位置を調整して FDD を取り付けネジ止め

▲FDD の下の 3.5 インチ・ドライブベイに HDD を差し込む

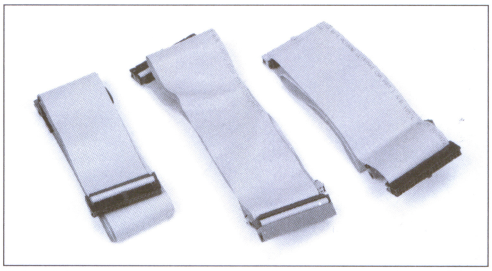


▲HDD はモータによる振動が出やすいので，とくにネジ止めは確実に行おう

5 電源&IDEケーブル類の配線→組み立て完了

DOS/V機製作の最終作業が，このケーブル配線です。IDE·FDDケーブル，電源ケーブル，さらにフロントパネル・ケーブルなど種類や本数が多く，動作しないなどのトラブルの原因になりやすいため，慎重な作業が求められます。ただし，フロントパネル・ケーブルを除けば，コネクタ形状がそれぞれ決められているので，接続端子を事前に確認しておけば戸惑うことはないはずです。

問題は，マザーボードへのフロントパネル・ケーブルの接続です。端子形状にピン数以外の区別がなく細かな作業になるため，どこに差せばよいのか，ベテランでさえ困ってしまうこともあります。マザーボードの付属マニュアルで確認しながら行うことが唯一の方法といえます。なお，Socket 478タイプのマザーボードは，12V補助電源端子が設けられていますので，忘れずに電源ケーブルを繋いでおきましょう。（文・福場龍夫）

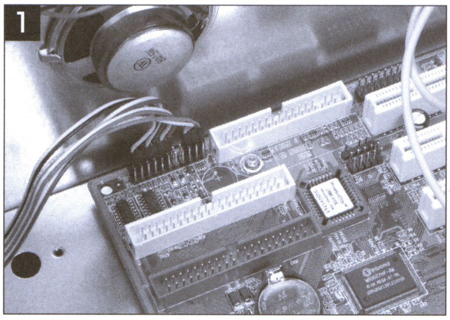


製品付属のフラットケーブル。左がFDD接続用で，他の2本はIDEドライブ接続用。ケーブルの線数やコネクタ形状が異なる

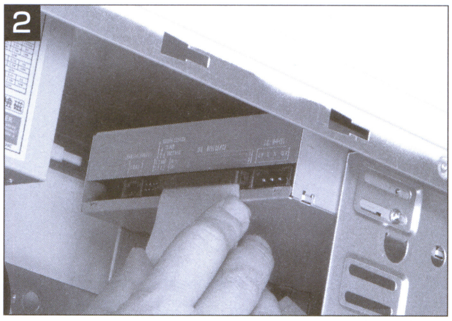
よくあるトラブルと対策法

『電源が入らない』『ドライブを認識しない』『パソコンの動作が不安定』
●対策：マザーボードやドライブへの電源ケーブル，FDDやIDEケーブルの接続を確認。『パソコンの動作が不安定』は，CPU強化電源ケーブルをチェック

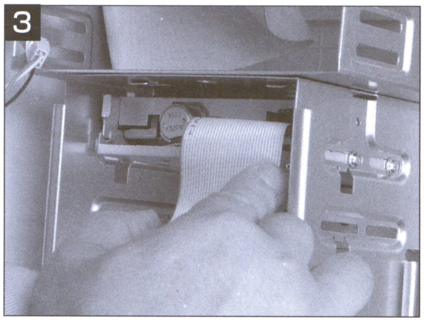
ケーブル配線



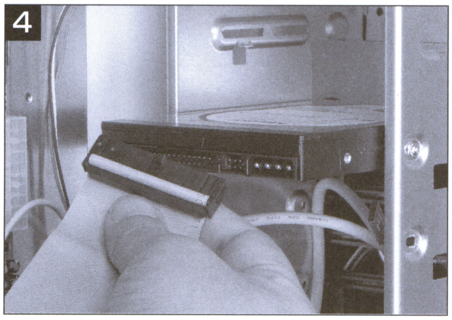
1 ▲フロントパネル・ケーブルの接続方法は，付属マニュアルの解説とおりに行う



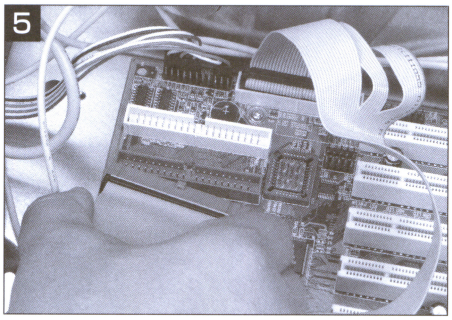
2 ▲CD(DVD)-ROMドライブのIDE端子へIDEケーブルを接続。差し込み方向に注意



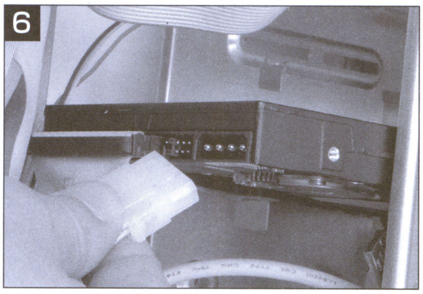
3 ▲FDDへFDDケーブルを接続。こちらも差し込み方向に注意



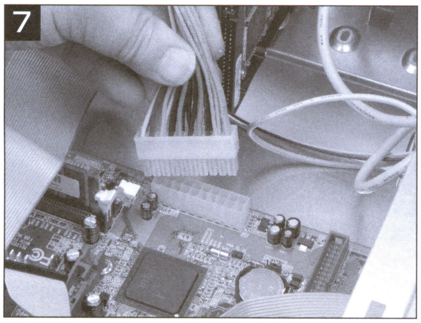
4 ▲IDEケーブルをHDDのIDE端子に接続。ケーブル端子の凸部分を上側にして差し込む



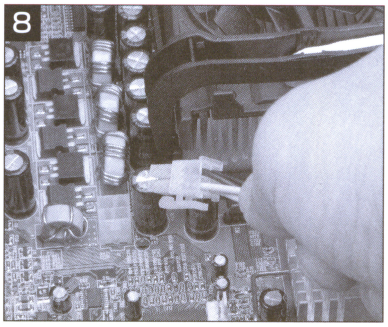
5 ▲IDEおよびFDDケーブルをマザーボードの各端子に接続



6 ▲電源ケーブルを各ドライブの電源端子に接続。FDDはコネクタ形状が異なる



7 ▲マザーボードの電源端子に電源ケーブルを接続する



8 ▲CPUのそばの12V補助電源端子に4ピン電源ケーブルを忘れずに接続

9 ◀DOS/Vマシン組み立て完了後，ウインドウズXPのインストールを行う

※PCディスプレイは別売

フリーソフト

HSP de ネットワークゲーム！

Hot Soup Processer

講師:うすあじ

WEBサイト「HSPセンター」を運営するHSPファンの一人。毎日テレビを見る男。

第2回 HSP/O2でジャンケンゲームを作る

こんにちは，うすあじです。

今回はHSP/O2使ったネットワークゲームの作り方を勉強していきましょう。

インストール

はじめにHSP/O2を使う準備をします。

(1)HSP本体をインストールする

O2を使うためには，2.6β13以降のバージョンをインストールしてください。

インストールの方法は**HSP入門コーナー**をご覧ください。すでにインストールしている人は，そのままでかまいません。

(2)HSP/O2をダウンロードする

O2のサイトからHSP/O2をダウンロードします。

http://o2.zener.co.jp/

ファイルを解凍すると，いくつかのフォルダとファイルが現われます。

(3)HSP/O2のファイルをコピーする

a.解凍したcommonフォルダに入っている二つのファイル，**o2.as**と**sephira.as**を，HSPをインストールしたフォルダ中のcommonフォルダにコピーします。

b.解凍したファイル**sephira.dll**をHSPをインストールしたフォルダにコピーします。これで，HSP/O2が使えるようになりました。

テスト

HSP/O2が使えるようになったか確認するために，同梱のサンプルを実行してみましょう。

・ローカルサーバーを動かす

サンプルをテストするために，テスト用のローカルサーバー**l_server.exe**を起動します。これがゲームサーバーの代わりをします。開発中には，このサーバーを使ってスクリプトのテストをします。

・サンプルを動かす

sampleフォルダの**click.as**を動かしてみましょう。二つのスクリプト間での通信を試す必要があるので，スクリプトを2回起動する必要があります。一つ目に起動したスクリプトに，ユーザーIDを入力します。テストなので，IDは適当に文字を入れればOKです。パスワードは要りません。これでロビーに入ります。

次に**Entry**ボタンを押すと**ROOM**が作られます。もう一つ起動した別のスクリプトにユーザーIDを入力します。ロビーに入ると，さっき作った**ROOM**が表示されます。その**ROOM**の文字をクリックすると**click.as**の本体が始まります。

click.asは二つのスクリプトで，マウスを順にクリックしていくものです。**sample**フォルダにあるスクリプトはHSP/O2でゲームを作る場合の参考になりますので，一度見ておくことをお勧めします。

ジャンケンゲーム

次にジャンケンゲームを紹介します。

○HSP/O2用のスクリプト

このスクリプト（**リスト**）は前半がHSP/O2のための準備で，後半がゲーム本体です。HSP/O2にアクセスするための命令を見ていきましょう。

・o2init

o2モジュールを初期化します。

最初に1度実行すればOKです。

・o2start

o2サーバーにログインする処理です。

ジャンケンゲームではパラメータp2に「-2」を指定していますが，これは2人用ゲームで，ローカルサーバーで実行するという指定です。値にマイナスを指定すると，ローカルサーバーでの実行になります。

・o2lobby

o2ロビーを初期化します。ロビーとはユーザー同士が待ち合わせをするための場所です。

・o2lchat

ロビーでチャットを行う処理です。

・o2wait

o2に対応したwait処理です。

・o2game

o2のゲームスクリーンを初期化します。

なんだか，沢山の「o2ナントカ」という命令がありますね。しかし，ゲームが始まるまでの部分は共通ですから，サンプルスクリプトを参考にしてください。

○ジャンケンゲーム本体

・ジャンケンとは

2人でジャンケンをすると

①2人のプレイヤーが,「グー」「チョキ」「パー」のどれかを同時に出す。

②双方の出した手(出したもの)を判定する。

③勝ち負けかあいこになる。

という流れになります。

「同時に出す」ということが，相手の

《リスト》ジャンケンゲームのソーススクリプト

```
;
;		ジャンケン
;
#include	"o2.as"
*main
		gosub *apinit
		o2init
		o2start "",-2				;ローカルサーバーで2
人用
*main1
		; ロビーを開始
		o2lobby
		objsize 80,18
		pos 414,410:button "exit",*btnexit
		onexit *btnexit
*main2
		; ロビー内のループ
		o2lchat
		if o2lchat_stat : goto *main3
		o2wait 30
		goto *main2
*btnexit
		; ロビーでexitボタンを押した
		o2lexit
		end
*errtrap
		; 致命的なエラーが起こった場合
		o2error
		end
*main3
		; ロビー終了
		if o2lchat_stat<0 : goto *errtrap
*main4
		; ゲームを開始
		o2game 4	; ゲームモード開始(チャット行数を指定)
		objsize 80,18
		pos 414,410:button "exit",*gcexit
		onexit *gcexit2
		gosub *ginit
		goto *main5
*gcexit
		; ゲームを終了(ロビーに戻る)
		gosub *gterm
		o2gexit
		goto *main1
*gcexit2
		; ゲームを終了
		gosub *gterm
		o2gexit
		end
*main5
		; ゲーム内のループ
		o2gcls				; ゲームエリアをクリ
ア
		gosub *gmain
		o2gupdate		; ゲームエリアを再描画
		o2wait 30
		o2gchat				; チャット機能
		if o2gchat_stat : goto *main6
		goto *main5
*main6
		if o2gchat_stat>0 : goto *gcexit
		goto *errtrap

;------------------------------------------------------------
		;		ここからゲーム固有のスクリプトになります
;------------------------------------------------------------
*apinit
		;		全体初期化のサブルーチン(最初の1回だけ呼ば
れます)
		;
		title "02 jyanken"			; タイトル名
		return
*ginit
		;		ゲーム初期化のサブルーチン(ゲーム開始時1回
だけ呼ばれます)
		;ジャンケンテーブル
		dim jtable,20
		;me	1,1,1,2,2,2,3,3,3  gu<1> choki<2> pa<3>
		;you	1,2,3,1,2,3,1,2,3
		jtable.0=3,1,2,2,3,1,1,2,3 ;kachi<1> make<2> aiko<3>

		sdim jname,10,5
		jname.1="グー"
		jname.2="チョキ"
		jname.3="パー"

		sdim kmesdt,10,5
		kmesdt.1="勝ち"
		kmesdt.2="負け"
		kmesdt.3="あいこ"

		user0=""
		user1=""
		seNameGet user0, 0
		seNameGet user1, 1
		myid=stat
		jyan=0 ;game mode
		return
*gterm
		;		終了処理のサブルーチン(最後の1回だけ呼ばれます)
		;
		return
*gmain
		;		毎フレームのサブルーチン(ゲーム中30msごとに呼ばれます)
		;
		seUserStat myid^1				; 相手がオンラインかチェック
		if stat=0 : o2gchat_stat=1 : return		; 相手がオフラインになった時

		if jyan=1:goto *jyan_mode1
		if jyan=2:goto *jyan_mode2

		; jyan_mode0
		font "MS ゴシック",32,16
		color 0,0,0:pos 180,80
		mes "じゃ～～んけ～～ん！"
		mes " "
		mes "←	↑	→"
		mes "グー チョキ パー"
		stick key,15,1
		myte=0
		if key&1:myte =1
		if key&2:myte =2
		if key&4:myte =3
		if myte :a=myid^1 : seDataPut myte,a:jyan=1
		return
*jyan_mode1
		font "MS ゴシック",32,16
		color 0,0,0:pos 200,80
		pos 180,80
		mes "ぽん！"
		mes jname.myte
		mes "wait"
		seDataStat
		if stat > 0 {
			seDataGet data
			jyan=2:jyan_ct=100
		}
		return
*jyan_mode2
		color 0,0,0:pos 180,80
		font "MS ゴシック",32,16
		k=(myte-1)*3+(data-1)
		kekka=jtable.k
		kmes=kmesdt.kekka
		mes "あなたは	"+jname.myte
		mes "相手は	"+jname.data
		mes "結果は	"+kmes
		jyan_ct--
		stick key,15,1
		if jyan_ct=0 {
			jyan=0
		}
		return
```

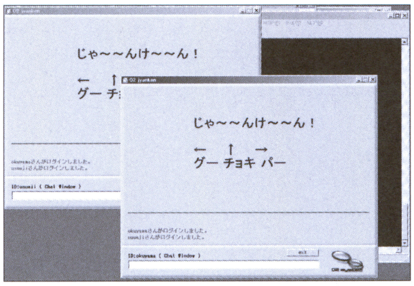
見えないネット上のゲームではむずかしいことです。今回は片方のプレイヤーの入力が終わっても相手側には表示しない，という仕掛けでジャンケンが成立するようになっています。

・モード

ジャンケンゲームには三つの状態があります。現在の状態(モード)は変数jyanに入っています。

モード0：

プレイヤーのジャンケンの入力待ち。カーソルキーで「グー」「チョキ」「パー」を入力します。入力したデータは，seDataPut命令で送信します（**写真1**）。



《写真1》プレイヤーのジャンケンの入力待ち

モード1：

相手の入力待ち。入力が終わるまでは，こちらの手を表示。 seDataStat命令でデータを確認して，seDataGet命令でデータを受け取ります（**写真2**）。

《写真2》相手の入力待ち

モード2：

結果を表示する。自分の手と相手の手は数値になっています。その組み合わせの勝敗が配列jtableに入っています。

「グー」「チョキ」「パー」の順でそれぞ

コラム

HSP入門コーナー

今月の課題

グラフィックを使ってみよう

HSPでグラフィックを操作する基本的な方法を見ていきましょう。

ウィンドウの初期化

これからグラフィックを表示しようというときにはウィンドウを初期化する必要があります。なにも指定しないでも表示することはできますが，その場合はウィンドウのサイズなどが初期値になっています。

ウィンドウを初期化する **screen**

例）screen 0,300,300

・ＩＤの割り当てかた

最初のウィンドウがID=0です。

次にウィンドウを作るときには，ID=2から割り当てます。

ID=1はHSPの内部で使用する特殊なウィンドウになっていますので使用できません。

バッファの初期化

screen命令で初期化したウィンドウと同じような動きで，画面上に見えないウィンドウが作れます。事前にグラフィックを読み込んで準備しておき，screenで初期化したウィンドウにコピーするときに便利です。

演劇で喩えるなら，スクリーンは舞台，バッファは楽屋といった感じでしょうか。準備する場所があると，舞台上での動きが順調に行くものです。

バッファを初期化する **buffer**

例）buffer 0,300,300

・スクリーンIDについて

screenで指定するIDとbufferで指定するIDには同じものを使います。たとえば，screen IDの2で初期化してしまうと，次にbuffer命令で初期化するのはID=3を使うことになります。

コピー命令などの指定でウィンドウIDを指定するときに「バッファの2」などと指定するわけでなく，スクリーンIDのみを指定するからです。

グラフィックを描くのは

color,boxf,line,pset

(1)色を設定　color命令

これから画面に何か描くぞ! というときに，はじめに描画色を設定します。描画色はRGBの三つのパラメータで，0-255の256段階のRGB分を設定します。

描画色を設定する **color**

例）color 0,255,0

例では「R=0,G=255,B=0」に指定しています。

(2)四角を描画　boxf命令

指定した座標に四角を描きます。p1,p2で左上の座標，p3,p4で右下の座標を指定します。このような二つセットの座標の場合はx座標，y座標の順に指定します。

四角を描画する**boxf**

例）boxf 100,100,200,200

(3)線を描画　line命令

指定した座標に線を描きます。p1,p2で一つ目の座標，p3,p4で二つ目の座標を指定します。その2点間の直線を描画します。

線を描画する**line**

例）line 100,100,200,200

(4)点を描画　pset命令

指定した座標に点を描きます。p1,p2に描画する座標を指定します。

点を描画する**pset**

例）pset 50,50

グラフィックを消すには

ウィンドウに何か描いたら，次に消す命令を紹介します。

・画面クリア1

一番簡単なのはcls命令です。cls命令は画面上にある，ボタンなどのオブジェクトも消します。簡単ですが，自由に画面の色が指定できないなどの欠点もあります。

画面をクリアする**cls**

例）cls

・画面クリア2

次にboxf命令を使った方法です。コンピュータ上で塗りつぶすと，完に消すのと同じ効果があります。

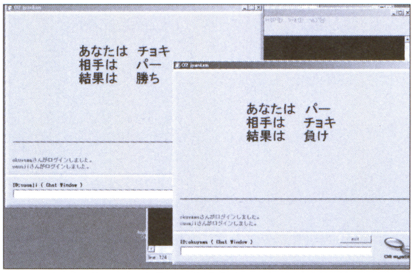
れ，自分が「グー」で相手が「グー」のとき，相手が「パー」のとき，という感じで勝敗のデータが入っています。データが1なら勝ち，2なら負け，3ならあいこです。

判定を全部if文で作ってもかまわないのですが，リストが長くなるので配列に入れました。

少し分かりにくいですが，スクリプトをよくみてください。結果を表示したらカウンタを回して，一定時間が過ぎたらモードを0に戻します（**写真3**）。

通常のゲームとは違い，少し特殊な作りになっています。

ゲーム中に30msに1回，＊gmainが呼ばれ，モードに応じた動作をします。動



《写真3》結果の表示画面

作が終わるとループに戻り，画面を更新したり，チャットの処理をします。

また，二つのスクリプト間でのやりとりは頭が混乱すると思います。図に書いたりしながら，じっくり作るといいでしょう。

次回は，ジャンケンスクリプトに手を加えていきます。

O2はonion softwareと（有）ツェナワークスが共同で提供しているサービスです。研究とネットゲーム作成環境の普及のために，一般に無料でサービスしています。

■フリーソフトHSPについては，本誌2002年1月号から7月号『HSP完全マスター楽々3Dプログラミング』で3Dゲームの記事を連載しています。

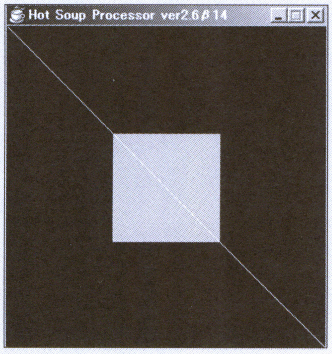
画面をクリアする**boxf**

例）color 0,0,0:boxf

boxfはパラメータを指定しないと，ウィンドウサイズの四角を描画します。

さて，ここまでの命令を集めてスクリプトを作りました。

```
screen 0,300,300,0
cls 4
color 0,255,0
boxf 100,100,200,200
color 255,255,255
line 0,0,300,300
pset 100,50
pset 100,100
stop
```



▲gra_samp.as

今月のまとめ

・ウィンドウの初期化
　screen,buffer
・グラフィックを描くのは
　boxf,line,pset
・グラフィックを消すには
　cls,boxf

コラム　HSPちょこっと入門

☆ダウンロードして使うまで☆

(1)インターネットからダウンロード！

HSP本体を手に入れよう。HSPはインターネット上のonion softwareのサイトからダウンロードできる。

http://www.onionsoft.net/hsp/

(2)圧縮ファイルを解凍！

ダウンロードしたファイルを解凍する。解凍したファイルを「c:¥hsp」などのディレクトリにＨＳＰをコピーしよう。

(3)準備する

HSPを使い始めるには，ファイル表示の設定を変更をします。この変更をしないと，拡張子が「dll」のファイルを見ることができません。

・「拡張子をすべて表示する」のチェックをつける。
・「登録された拡張子を表示しない」のチェックをはずす。

(4)使ってみる

コーヒーポットのアイコン「hsed2.exe」をダブルクリックして動かします。ウィンドウの黒い部分にスクリプトを入力して，F5キーを押すとスクリプトを実行します。

では，スクリプトを入力してみましょう。

```
mes "Hello HSP!"
stop
```

最初は，すべて半角の英数文字で入力してください。入力し終わったらF5キーを押してみましょう。

Hello HSP!

うまくいくとメッセージが表示されます。あなたのコンピュータでHSPが動きました。

これからは，付属のドキュメントやWEBサイト，本誌に投稿されているスクリプトなどを参考にしながらHSPを楽しんでください。

.NET GAME PROGRAMMING ドットネットゲームプログラミング

.NETでゲームを作ろう!

ヒットゲーム[前編]

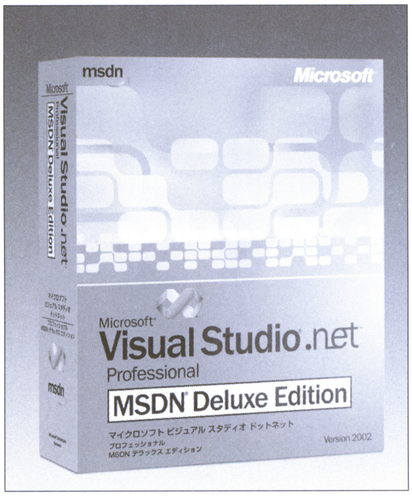
4

赤坂　玲音

.NET 開発環境

Microsoft .NET (ドットネット)とは，Web やシステムのサービスを統合し，従来よりも高度なソリューションを実現するために Microsoftが開発した新しいプラットフォームです。.NET は Java のように仮想マシンで動作するため，理論上システムやコンピュータに依存することはありません。

そのため，ネットワークに強く，強力な標準クラスライブラリも用意されています。従来の Windows アプリケーションの開発で用いられていたWin32 APIや MFC による開発よりも生産性が向上し，ソフトウェアの再利用が容易になります。ここでは，.NET アプリケーションの開発方法をC#言語とVisualBasic .NETを使って，ゲームを作りながら解説していきます。.NET アプリケーションの開発には，Microsoft が提供する .NET開発環境を使ってプログラムします。.NET開発環境は Windows 95/98/Me にインストールすることはできないので，注意してください。



《写真1》Visual Studio .NET Professional version 2002 MSDN Deluxe Edition

通常，.NET アプリケーションの開発には Microsoft Visual Studio .NET (以下 VS.NET) を使います。

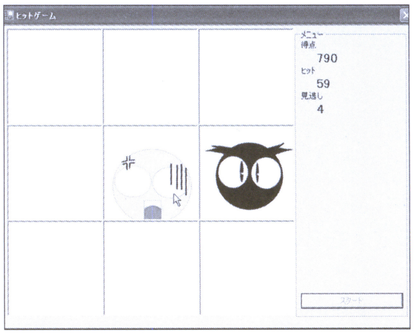
VS.NET は，Professional(写真1) の場合，メーカー指定小売価格で158,000円と高価ですが，大規模な開発に適していて，コンパイルだけではなく開発管理やコードの自動生成など多くの面で開発をサポートしてくれるので，生産性が向上し保守も容易になるでしょう。学生や初心者には，Standard 版が用意されています。この場合は C# や VB .NET などのコンパイラが個別でしかサポートされませんが，価格も12,800円と，初心者，入門者に適した総合開発環境です。また，.NET Framework SDK と呼ばれる，基本的なツールを備えた開発キッドを無料で手に入れることもできます。.NET Framework SDK は，Microsoft のサイト http://www.microsoft.com/japan/msdn/netframework/ から無償でダウンロードすることができるので，もっとも安上がりに.NET アプリケーションを作成する手段です。ただし，.NET Framework SDKは総合開発環境ではないため，コンパイルはコマンドラインから行う必要があります。Framework SDK は，小規模なライブラリやプログラムの作成など，.NET の体験や実験に適しているといえるでしょう。

ゲーム仕様

今回は，コントロールにキャラクターが表示されている状態でマウスをクリックすると得点が加算されていく「ヒットゲーム」(写真2) を作ります。いわゆる「もぐらたたき」です。

このゲームは，スタートボタンを押すとゲームが起動し，コントロールに一定のタイミングでキャラクターが表示されます。キャラクターが表示されている状態でマウスの左ボタンをクリックすると，キャラクターの絵が変わって叩かれたことをアピールします。

また，キャラクターが表示されてから一定の時間が経過するとキャラクターの



《写真2》ヒットゲーム

絵が消えます。つまり見逃しです。コントロールに表示されるキャラクターの画像の数は複数あってもかまいません。キャラクターが叩かれたり，見逃された場合は得点が変化します。

このようなゲームが終了するタイミングとして考えられるのは，一定の数が表示されるか，一定の時間が経過した場合でしょう。今回は，一定の数が表示された場合に終了するものとします。

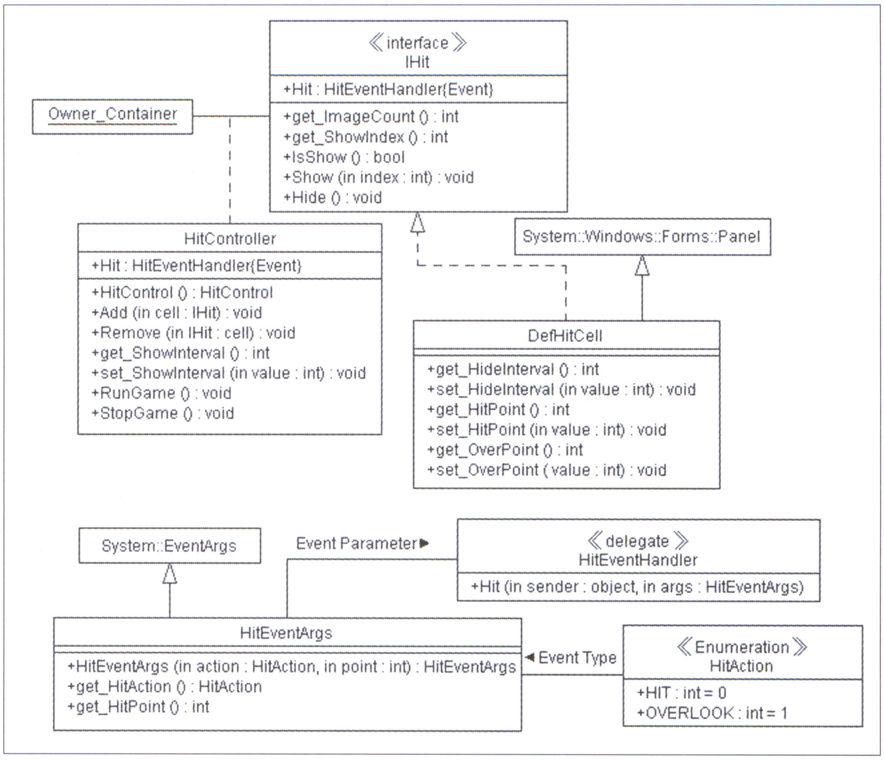
設計

このゲームの開発意図は，オブジェクト指向による設計技法の基本を解説するためです。前回までは，Windowsコントロールを使ったプログラミングの基礎を学ぶためにスロットマシンを作成しましたが，スロットに画像を表示したり，本物のスロットのようにクルクル回転させることはできませんでした。こうした処理を実現させたいと思った場合，コード全体を書き直さなければならないでしょう。

実際のゲーム開発の現場では，そのようなプログラムを作るわけにはいきません。ゲームのようなエンターテイメント商品は，開発中の段階に湧き出たアイデアも導入できなければならないのです。クリエイターの突然の提案を導入するために，プログラム全体を書き直すのは効率的ではありません。これはゲームばかりではなく，多くのソフトウェアの開発現場に共通していて，ある日突然，誰かが画期的なアイデアを提案するかもしれないのです。

このように，プログラムを頻繁に拡張する可能性がある場合は，設計に注目しなければなりません。現在，肥大化し続ける市場と技術に対応するには，常に拡張可能なプログラムを作らなければならないと言っても過言ではないでしょう。.NETの機能とオブジェクト指向を生かせば，この問題をスマートに解決することができます。

プラグイン可能な柔軟なシステムを建築する場合は，インターフェース→抽象クラス→デフォルトクラスという順番に開発を進めます。インターフェースは，何らかのコントロールやシステムに設定するオブジェクトが実装しなければならない基本的なメソッドを宣言し，抽象クラスは，インターフェースで定められている仕様に準拠した機能を提供し，その他の便利なメソッドなども追加されます。



《第1図》ヒットゲームの設計図

デフォルトクラスは，デフォルトで設定されるべきシステムが提供する実装となります。通常，開発者は抽象クラスを特化することによって，最小の労力で目的の拡張を実現することができます。しかし，システムはインターフェースを通して機能を呼び出しているため，開発者はインターフェースを直接実装した独自のクラスを設定することもできるでしょう。

オブジェクト指向による設計論を本格的に解説するとなれば，分厚い一冊の本が出来上がってしまいます。この場では，柔軟なコンポーネント・システムを開発するための基本である「インターフェースを操作する」を念頭に開発を進めましょう。

第1図は，今回作成するヒットゲームをUMLの静的構造図（クラス図）で表現したものです。各クラスについての詳細は，これから少しずつ解説していくので，開発の要約として目を通してください。作成するクラスの数から考えても，それほど複雑なものではありません。

まず，**写真2**のキャラクターを表示するためのコントロールに着目しましょう。この場では，このコントロールを「ヒット・セルコントロール」と呼ぶことにします。

ゲームの完成品だけを見れば，ヒット・セルコントロールはイメージを描画するコントロールを作成するだけのように見えますが，再利用可能な柔軟なアプリケーション開発の場合は，実体だけを作ればよいというものではありません。まず，すべてのヒット・セルコントロールが実装しなければならないメソッドを宣言したインターフェイスIHitを作ります。

そして，デフォルトクラスとなるDefHitCellクラスでIHitを実装します。**写真2**でキャラクター表示しているコントロールはDefHitCellクラスのものです。IHitクラスとDefHitCellの間に抽象クラスを設けるべきですが，設計に徹底するとコードが冗長になるため，雑誌というメディアであることを考慮して省略します。

HitControllerは，ヒット・セルコントロールを追加する親コンテナと，複数個のヒット・セルコントロールの仲介者となる関係クラスです。複数のヒット・セルコントロールの制御をコンテナが管理する場合，他のコンテナに移植する事態を考えると柔軟ではないことがわかる

でしょう。

そこで，複数個のヒット・セルコントロールをまとめて管理し，ゲームの流れを制御してくれるコントローラが必要なのです。この関係クラスを，「ヒット・コントローラ」と呼ぶことにします。

HitEventHandler はインターフェースです。このインターフェースは，ヒット・セルコントロールのキャラクターがクリックされたり，見逃された時にコールバックするメソッドを宣言しています。.NETであれば，これはデリゲートとして扱うことができますが，設計図の段階では特定の言語に依存しないように記述することが望ましいので，インターフェースとして扱っています。コールバック機能をもたない言語を用いて作る場合は，インターフェースのメソッドを呼び出すことでイベントを通知することができます。

HitEventHandlerデリゲートは，イベント情報を格納したHitEventArgsクラスのオブジェクトを受け取ります。HitAction 列挙型は，ヒット・イベントの種類を表す値を提供しています。

以上が，今回作成するゲームの基本的な設計概念です。この設計図も完全なものではなく，いくつかの問題点を含んでいます。例えば，ヒット・コントローラが提供する機能が抽象化されていないのは致命的な弱点です。 このような問題を解決する方法も含めて，解説を進めていきたいと思います。

IHit インターフェース

まず最初に，IHit インターフェースを宣言し，基本的な枠組みを作ってしまいましょう。IHitインターフェースは，**第1表**のようなビヘイビア（操作特性）を定めるものとします。

《第1表》IHit インターフェースの仕様

プロパティ	解説
int ImageCount {get;}	表示可能なキャラクター数を返す
int ShowIndex { get; }	現在表示しているイメージのインデックスを返す 非表示状態であれば -1 を返す
bool IsShow { get; }	キャラクターが表示されていれば true を，そうでなければ false を返す
メソッド	
void Show(int index)	index で指定したキャラクターを表示する
void Hide( )	キャラクターを隠す
イベント	
event HitEventHandler Hit	表示中のキャラクターをマウスでクリックしたり， 一定時間の経過により見逃された時に発生するヒット・イベント

《コード1》IHit インターフェース

```
public interface IHit
{
        int ImageCount { get; }        //表示するイメージの数
        int ShowIndex { get; }         //現在表示しているイメージのインデックス
        bool IsShow { get; }           //イメージを表示しているのであれば true
        void Show(int index);          //指定イメージのインデックス表示
        void Hide();                   //イメージを隠す
        event HitEventHandler Hit;
}
```

《コード2》IHitインターフェースVB.NET

```
Public Interface IHit
    ReadOnly Property ImageCount() As Int32 '表示するイメージの数
    ReadOnly Property ShowIndex() As Int32  '現在表示しているイメージのインデックス
    ReadOnly Property IsShow() As Boolean   'イメージを表示しているのであれば true
    Sub Show(ByVal index As Int32)          '指定イメージのインデックス表示
    Sub Hide()                              'イメージを隠す
    Event Hit As HitEventHandler
End Interface
```

IHit インターフェースは，直接イメージデータをやり取りするようなことはありません。**写真2**のように，完成したゲームのヒット・セルコントロールは画像ファイルを読み込んで，イメージをコントロールに描画していますが，他の実装では必ずしもイメージを使うとは限らないからです。

イメージを使わずにボタンのようなものでゲームが作られるかもしれませんし，GDI+で図を描画するかもしれないのです。そこで，IHitインターフェースは表示するキャラクターをインデックスで指定します。ImageCount プロパティやShow() メソッドはそのためのものです。

ヒット・イベントはHitイベントを呼び出すことで発生します。イベント発生のタイミングは実装に委ねられているため，必ずしもマウスのクリックでヒット・イベントが発生するわけではありません。なにより，IHitは実装がWindows コントロールであることを強制していないのです。

ヒット・イベント

ヒット・イベントは，ヒット・セルコントロールで表示されているキャラクターが叩かれたり，見逃された時に発生するイベントです。イベントの発生を受けるには，HitEventHandler デリゲート型のインスタンスを作成しなければなりません。HitEventHandlerは，.NETのイベント規則に従って第1パラメータにイベントを発生させたオブジェクトを，第2パラメータにイベント引数クラスを受け取ります。ヒット・イベントの情報はHitEventArgs クラスが担当します。

また，HitEventArgsクラスはSystem.EventArgsクラスを継承し，ヒット・イベントの種類を返す HitAction プロパティと，加算される点数を返すHitPoint プロパティを提供します。System.EventArgsはイベントデータを格納するクラスの基本クラスです。イベントの種類はHitAction列挙型で定義されています。 HitAction.HITは叩かれた（ヒット）ことを表し，HitAction.OVERLOOK は見逃されたことを表すものとします。

HitEventArgsクラスのコンストラクタ

《コード3》ヒット・イベント C#

```
//ヒットイベントの種類を表した列挙型
public enum HitAction
{
                HIT ,               //叩いた
                OVERLOOK            //見逃した
}
//ヒットイベントの情報を格納した引数クラス
public class HitEventArgs : EventArgs
{
                private HitAction action;
                private int point;

                public HitEventArgs(HitAction action , int point)
                {
                 this.action = action;
                 this.point = point;
                }
                public HitAction HitAction
                {
                 get { return action; }
                }
                public int HitPoint
                {
                 get { return point; }
                }
}

//ヒットイベントハンドラ
public delegate void HitEventHandler(Object sender , HitEventArgs args);
```

《コード4》ヒット・イベント VB .NET

```
'ヒットイベントの種類を表した列挙型
Public Enum HitAction
    HIT         '叩かれた
    OVERLOOK    '見逃した
End Enum

'ヒットイベントの情報を格納した引数クラス
Public Class HitEventArgs
    Inherits EventArgs

    Private action As HitAction
    Private point As Int32

    Public Sub New(ByVal action As HitAction, ByVal point As Int32)
        Me.action = action
        Me.point = point
    End Sub

    ReadOnly Property HitAction() As HitAction
        Get
            Return action
        End Get
    End Property

    ReadOnly Property HitPoint() As Int32
        Get
            Return point
        End Get
    End Property
End Class

'ヒットイベントハンドラ
Public Delegate Sub HitEventHandler(ByVal sender As Object, ByVal args As
HitEventArgs)
```

では，HitAction列挙型の値と，このイベントによって加算される値を渡します。もし，イベントによって点数が減点される場合は負数を指定します。

コード3，または**コード4**を見てわかるようにHitEventArgsクラスはコンストラクタで必要な値を受け取り，非公開のフィールドに保存します。イベントが発生すると，デリゲートのインスタンスがこれを受けることため，HitActionやHitPointプロパティからコンストラクタで設定された値を取得することができます。

ヒット・コントローラ

ヒット・コントローラは，ヒット・セルコントロールと，それを利用するクラス（コンテナ）との関係クラスです。ヒット・コントローラの役割は，複数個のヒット・セルコントロールを管理し，コンテナからの要求を実現するために存在します。**第1図**を見てもわかるように，ヒット・コントローラはIHitインターフェースを操作するため，実装は問題ではありません。

しかし，HitControllerには設計上いくつかの問題があります。**第2表**にHitControllerクラスの操作特性の説明がありますが，仕様が特化しすぎているのです。できることならば，キャラクターの表示/非表示を操作するアルゴリズムを抽象化し，プラグイン可能な柔軟性を確保するべきだと考えられます。そうすれば，難易度に応じて表示アルゴリズムを選択できる仕組みになるでしょう。

ですが，抽象化やモデル化を進めすぎるとコードが肥大化するため，ヒット・コントローラの抽象化は避けることにします。

ヒット・コントローラでまず重要なのがAdd()メソッドとRemove()メソッドです。これらのメソッドは，リストのようにヒット・セルコントロールの追加と削除を行う機能を提供します。Add()メソッドにIHitを渡せばコントローラの管理対象となりRemove()を呼び出すと管理対象から削除されます。管理対象となっているIHitは，HitControllerの

《第2表》HitController クラスの仕様

プロパティ	解説
int ShowInterval { get; set;}	ヒット・セルコントロールがキャラクターを表示する間隔をミリ秒単位で設定/取得する
メソッド	
void Add(IHit cell)	cell で指定したヒット・セルコントロールをコントローラに追加する すでに cell が管理対象となっている場合は何も行わない
void Remove(IHit cell)	cell で指定したヒット・セルコントロールをコントローラから削除する cell がコントローラに存在しない場合は何もしない
void RunGame()	ゲームを開始する
void StopGame()	表示中の全てのキャラクターを隠してゲームを終了する
イベント	
event HitEventHandler Hit	ヒット・セルコントロールで発生したイベントをスルーする

《第3表》ILIst インターフェース

プロパティ	解説
bool IsFixedSize { get; }	固定サイズかどうかを示す値を取得する
bool IsReadOnly { get; }	読み取り専用かどうかを示す値を取得する
object this[int index] { get; set; }	指定したインデックスにある要素を取得または設定するインデクサ
メソッド	
int Add(object value)	項目を追加する
void Clear()	すべての項目を削除する
bool Contains(object value)	特定の値が格納されているかどうかを判断する
int IndexOf(object value)	特定の項目のインデックスを判断する
void Insert(int index,object value)	指定した位置に項目を挿入する
void Remove(object value)	指定したオブジェクトと同一の，最初に出現したオブジェクトを削除する
void RemoveAt(int index)	指定したインデックスにある項目を削除する

RunGame() メソッドが呼び出されゲームが開始すると，ランダムにShow() メソッドが呼び出されるようになります。

可変個数のオブジェクトを扱う機能を自分で実現するのは少し面倒です。そこで，.NETの標準クラスライブラリが提供するSystem.Collections.ArrayList クラスを使いましょう。

```
[Serializable] public class ArrayList :
IList, ICollection, IEnumerable,
ICloneable
```

ArrayListクラスは System.Objectを継承し，いくつかのインターフェースを実装しています。 その中でも特に重要なのはSystem.Collections.IListインターフェースを実装しているということです。

```
public interface IList : ICollection,
IEnumerable
```

IList インターフェースは，まさに私たちが今求めているAdd()メソッドやRemove() メソッドを宣言しています。IList.Add()メソッドはオブジェクトをリストに追加し，IList.Remove() メソッドはオブジェクトをリストから削除します。また，指定したインデックスの項目を削除するIList.RemoveAt() や，項目にアクセスするためのインデクサなども宣言されています。

第3表は，IListインターフェースで宣言されているプロパティとメソッドです。これらの機能がArrayListクラスで実装されているため，HitController ではIHitの管理にこれを利用することにしましょう。このプログラムでは，ヒット・セルコントロールがコントローラに追加される順番は重要ではないので HitController は IListを実装しないものとします。

コード5や**コード6**のように，ArrayListクラスを使えば，簡単にオブジェクトの追加と解除処理を実現することができます。このような処理は，可変個数のオブジェクトを扱うプログラムで頻繁に使われるため，覚えておくと良いでしょう。

ヒット・コントローラのヒット・セルコントロールに，キャラクターを表示するように通知する手段には，前回のスロットゲームで学習したタイマーを使うことにしましょう。ShowIntervalプロパティで，コントローラに登録されている適当なIHit のShow()メソッドを呼び出す間隔を指定することができるようにし，RunGame() メソッドが呼び出されれば，タイマーを起動するように仕組みます。タイマーイベントが発生すれば，乱数を使ってIHitをランダムに選択して，さらにShow() で表示されるキャラクターインデックスも乱数で決定させましょう。

キャラクターが表示されれば，ヒット・セルコントロールは，近い将来ヒット・イベントを発生させることでしょう。イベントはIHitのインスタンスごとに登録してもかまいませんが，HitController.Hit が登録された IHit のヒット・イベントを一括して発生させるので，こちらを用いた方が効率的に行えます。 **コード5**，または**コード6**では，IHit が HitController.Add() によって追加される時に，イベントに HitControllercell_Hit メソッドを登録しています。IHit のインスタンスがヒット・イベントを発生させると，HitController.cell_Hitメソッドは何もせずに発生したイベントをスルーします。

基盤の完成

さて，ヒット・セルコントロールのインターフェースと，ヒット・コントローラが完成すれば，基本的な部品は完成したことになります。このように，それぞれの役割を分割し，抽象化を進めることによって柔軟性の高いシステムを建築することができます。あとは，IHitインターフェースを実装したヒット・セルコントロールの実体となるクラスを作成するだけですが，これは次回に解説します。

今回作成したプログラムで最も重要なのは，他の何らかの機能を提供するクラス群にアクセスする場合，直接実体を呼び出してはいけないということです。

HitController クラスは，ヒット・セルコントロールの実体ではなく，IHit インターフェースのみを扱っています。

このように，インターフェースや抽象クラスにのみアクセスすることによって，将来の拡張や開発中に発見されたバグに強いシステムを建築できるのです。オブジェクト指向において，一部を書き直すために全体を調整しなければならないようなプログラムは最低です。早くゲームを作って動かしてみたいプログラマーにとって，実体をすぐに書けないのは面倒に感じるかもしれませんが我慢しましょう。

《コード5》Add()とRemove()の実装C#

```
private System.Collections.ArrayList list =   new System.Collections.ArrayList();
public event HitEventHandler Hit;

public void Add(IHit cell) {
        if (list.Contains(cell)) return;

        list.Add(cell);
        cell.Hit += new HitEventHandler(cell_Hit);
}
public void Remove(IHit cell)
{
        if (list.Contains(cell))
        {
                cell.Hit -= new HitEventHandler(cell_Hit);
                list.Remove(cell);
        }
}

private void cell_Hit(object sender , HitEventArgs e)

        Hit(sender , e);
}
```

《コード6》Add()とRemove()の実装 VB.NET

```
Private list As System.Collections.ArrayList = New System.Collections.ArrayList()
Public Event Hit As HitEventHandler

Public Sub Add(ByVal cell As IHit)
    If list.Contains(cell) Then Return

    list.Add(cell)
    AddHandler cell.Hit, AddressOf Me.cell_Hit
End Sub

Public Sub Remove(ByVal cell As IHit)
    If list.Contains(cell) Then
        RemoveHandler cell.Hit, AddressOf cell_Hit
        list.Remove(cell)
    End If
End Sub

Private Sub cell_Hit(ByVal sender As Object, ByVal e As HitEventArgs)
    RaiseEvent Hit(sender, e)
End Sub
```

《コード7》ヒットゲームの骨組み C#

```
using System;
using System.Drawing;
using System.Collections;
using System.Windows.Forms;

//ヒットイベントの種類を表した列挙型
public enum HitAction
{
        HIT ,    //叩いた
        OVERLOOK //見逃した
}

//ヒットイベントの情報を格納した引数クラス
public class HitEventArgs : EventArgs
{
        private HitAction action;
        private int point;

        public HitEventArgs(HitAction action , int point)
        {
                this.action = action;
                this.point = point;
        }

        public HitAction HitAction
        {
                get { return action; }
        }

        public int HitPoint
        {
                get { return point; }
        }
}

//ヒットイベントハンドラ
public delegate void HitEventHandler(Object sender , HitEventArgs args);

// 全てのヒット・セルコントロールが実装しなければならないインターフェイス
public interface IHit
{
        int ImageCount { get; }            //表示するイメージの数
        int ShowIndex { get; }             //現在表示しているイメージのインデック
ス
        bool IsShow { get; }               //イメージを表示しているのであれば true
```

```
        void Show(int index);                //指定インデックスのイメージを表示
        void Hide();                         //イメージを隠す
        event HitEventHandler Hit;
}

//ヒット・コントローラ
public class HitController
{
        private System.Collections.ArrayList list =   new
System.Collections.ArrayList();
        private Timer timer1 = new Timer() ;
        private Random random1 = new Random();

        public event HitEventHandler Hit;

        public HitController()
        {
                timer1.Tick += new EventHandler(timer1_Tick);
                timer1.Interval = 1000;
        }

        public void Add(IHit cell) {
                if (list.Contains(cell)) return;

                list.Add(cell);
                cell.Hit += new HitEventHandler(cell_Hit);
        }

        public void Remove(IHit cell)
        {
                if (list.Contains(cell))
                {
                        cell.Hit -= new HitEventHandler(cell_Hit);
                        list.Remove(cell);
                }
        }

        public int ShowInterval
        {
                get { return timer1.Interval; }
                set { timer1.Interval = value; }
        }

        public void RunGame()
        {
                timer1.Start();
        }
        public void StopGame()
        {
                timer1.Stop();
                for(int i = 0 ; i < list.Count ; i++)
                        ((IHit)list[i]).Hide();
        }

        private void timer1_Tick(object sender , EventArgs e)
        {
                int cellIndex = random1.Next(-1 , list.Count);
                if (cellIndex == -1) return;
                else
                {
                        IHit hit = list[cellIndex] as IHit;
                        if (hit != null && !hit.IsShow)
                        {
                                int imageIndex = random1.Next(hit.ImageCount);
                                hit.Show(imageIndex);
                        }
                }
        }
        private void cell_Hit(object sender , HitEventArgs e)
        {
                Hit(sender , e);
        }
}
```

《コード8》ヒットゲームの骨組みVB.NET

```
Imports System
Imports System.Drawing
Imports System.Collections
Imports System.Windows.Forms

'ヒットイベントの種類を表した列挙型
Public Enum HitAction
    HIT         '叩かれた
    OVERLOOK    '見逃した
End Enum

'ヒットイベントの情報を格納した引数クラス
Public Class HitEventArgs
    Inherits EventArgs

    Private action As HitAction
```

```
    Private point As Int32

    Public Sub New(ByVal action As HitAction, ByVal point As Int32)
        Me.action = action
        Me.point = point
    End Sub

    ReadOnly Property HitAction() As HitAction
        Get
            Return action
        End Get
    End Property

    ReadOnly Property HitPoint() As Int32
        Get
            Return point
        End Get
    End Property
End Class

'ヒットイベントハンドラ
Public Delegate Sub HitEventHandler(ByVal sender As Object, ByVal args As
HitEventArgs)

Public Interface IHit
    ReadOnly Property ImageCount() As Int32 'イメージの総数
    ReadOnly Property ShowIndex() As Int32  '現在表示しているイメージのインデックス
    ReadOnly Property IsShow() As Boolean   'イメージを表示しているのであれば true
    Sub Show(ByVal index As Int32)          '指定インデックスのイメージを表示
    Sub Hide()                              'イメージを隠す
    Event Hit As HitEventHandler
End Interface

'ヒット・コントローラ
Public Class HitController
    Private list As System.Collections.ArrayList = New System.Collections.ArrayList()
    Private timer1 As Timer = New Timer()
    Private random1 As Random = New Random()

    Public Event Hit As HitEventHandler

    Public Sub New()
        AddHandler timer1.Tick, AddressOf timer1_Tick
        timer1.Interval = 1000
    End Sub

    Public Sub Add(ByVal cell As IHit)
        If list.Contains(cell) Then Return
        list.Add(cell)
        AddHandler cell.Hit, AddressOf Me.cell_Hit
    End Sub

    Public Sub Remove(ByVal cell As IHit)
        If list.Contains(cell) Then
            RemoveHandler cell.Hit, AddressOf cell_Hit
            list.Remove(cell)
        End If
    End Sub

    Property ShowInterval() As Int32
        Get
            Return timer1.Interval
        End Get
        Set(ByVal Value As Int32)
            timer1.Interval = Value
        End Set
    End Property

    Public Sub RunGame()
        timer1.Start()
    End Sub
    Public Sub StopGame()
        timer1.Stop()
        Dim i As Int32
        For i = 0 To list.Count - 1 Step 1
            CType(list(i), IHit).Hide()
        Next
    End Sub

    Private Sub timer1_Tick(ByVal sender As Object, ByVal e As EventArgs)
        Dim cellIndex As Int32 = random1.Next(-1, list.Count)
        If cellIndex = -1 Then
            Return
        Else
            Dim hit As IHit = CType(list(cellIndex), IHit)
            If Not hit Is Nothing And Not hit.IsShow Then
                Dim imageIndex As Int16 = random1.Next(hit.ImageCount)
                hit.Show(imageIndex)
            End If
        End If
    End Sub
    Private Sub cell_Hit(ByVal sender As Object, ByVal e As HitEventArgs)
        RaiseEvent Hit(sender, e)
    End Sub
End Class
```

オリジナルソフトが作れる！
持ち歩ける！PC用携帯端末

「PIECE」プログラミング

第7回 トランプゲーム「ブラックジャック」編

文：まかべひろし

101×65×17mm（縦×横×厚さ），92g（電池含む）
発売元：アクアプラス
価格：9800円（税別）
色：シルバー（初回限定），クリアブルー，クリアレッド，パールホワイト
http://www.piece-me.com/

スペック

CPU	EPSON S1C33209（32ビットRISC、24MHzで動作）
メモリ	SRAM 256Kbyte フラッシュRAM 512Kbyte
液晶	FSTN 白黒4階調 128×88ドット
音声出力	本体出力：モノラルスピーカー 外部出力：ヘッドホン端子
インターフェース	USB／赤外線
付属フォント	JIS第一・二水準漢字フォント(10×10ドット)
電源	単三電池×1
パソコン側環境	WindowsXP/2000/Me/98（要USB）

HANDBOOK Vol.2，大好評発売中！

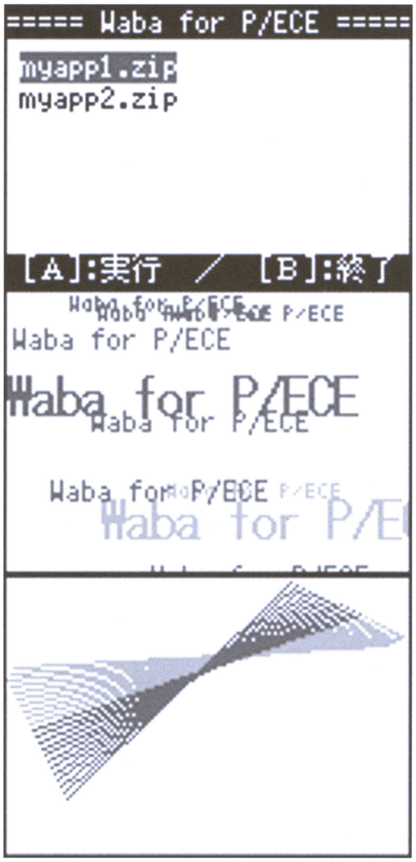
お待たせしました! ついについに電波新聞社の「P/ECE HANDBOOK Vol.2」が発売となりました！その内容は，専門誌らしくPIECE一色。これでもか，というほど，PIECEの魅力的な世界を堪能できます。

ゲームで遊びたい人は3本の「HANDBOOKオリジナルゲーム」，ハードウェアに興味がある人は「ハードウェアスーパーテクニック」，オリジナルゲームのアルゴリズム解説もあるので，より高度なプログラムを知りたい人も大満足。また，PIECEを使って便利に生活する記事も満載していますので，ぜひぜひ読んで，できればお買いあげくださいねっ。

HANDBOOK Vol.2プロジェクトは発売を迎えて，無事終了したけれど，ベーマガ本誌の連載は，まだまだ続いていくのです！毎月ソフト1本を目標に，どんどん作っていくので期待していてくださいね！

PIECEでJAVAが動いた！？

携帯端末でのプログラミング環境といえば，携帯電話iモードなどに搭載されている「JAVA」が有名ですが，なんと，我らがPIECEでもJAVAプログラムが動く日がやってきました！　正確には，JAVAと同じ文法仕様をもつ，いわば携帯端末用JAVAとも言える「Waba」という環境。Wabaには，メモリの少ないマシンでも動くように，不要な機能や，メモリを大量に使用する機能が搭載されていないため，PIECEでも動いてしまうのです。



▲Waba for PIECEの実行画面

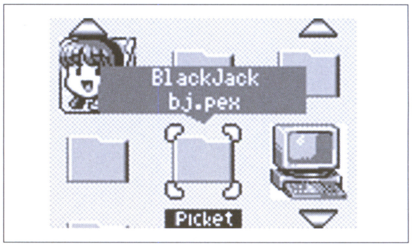
開発したのは，本連載でも「消費電力軽減」などでおなじみ，PIECE研究室のnsawa氏。現在，Waba仕様の基本的な部分をサポートしたものを公開されています。まだまだプログラム環境としては不十分ではありますが，今後の実装により，「PIECEの上でJAVAプログラム」が期待できそうです。C言語や，PIECE独自のAPIに抵抗感がある人は，試してみてはいかがでしょうか。

ホント，PIECEには，まだまだ隠された力があるようです。奥が深い。

●PIECE研究室
http://north.hokkai.net/~nsawa/
(Waba for PIECEの話題は，2002年11月29日が初出)

おしゃべりブラックジャック！

2002年12月号に掲載されたトランプゲーム「スピード」，意外に好評なようでしたので，調子に乗って今回もトランプゲームを用意しました。リストを掲載することが大前提ですから，それほど大規模なものはできません。そこで，カジ



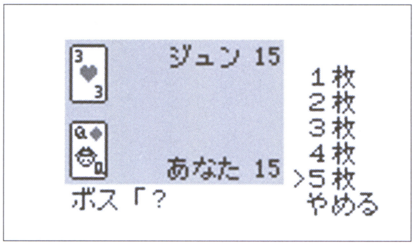
▲「ブラックジャック」タイトル画面

ノなどでもおなじみ，合計「21」を目指していくゲーム「ブラックジャック」を作ってみました。

しかし，ただのブラックジャックでは面白くありません。今回は，相手にいろいろおしゃべりをさせてみました。ところで，おしゃべりをするブラックジャックゲームといえば，非常に有名な，ボクにとっても思い出ぶかいゲームがあります。今回は，その有名なゲームを模倣する形で作ってみました。見覚えのある人は，ニヤリとしてくださいな。

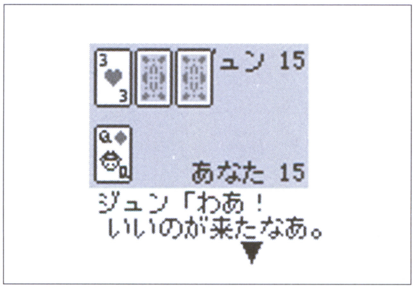
遊び方ですが，ブラックジャックは，手札の合計を相手よりも多くすることが目的のゲームです。ただし，合計が21を超えたら即負け。そのほかにも，以下のようなルールがあります。

・絵札は10と計算する
・Aは1，もしくは11と計算する



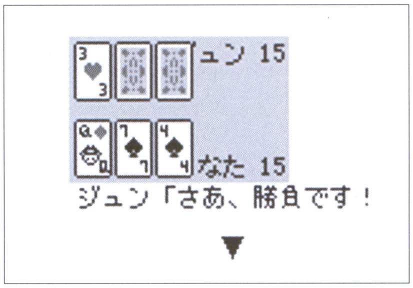
▲カードが1枚ずつ配られる

一番最初，相手と自分に，カードが1枚ずつ配られます。この時点で，賭けチップ数を決定します。最大5枚まで賭けられますので，相手と自分のカードを見て，好きな枚数を賭けて下さい。



▲さあ，どう勝負するか？

その後，相手のカードは見えなくなります。このため，相手の合計がいくつになるかはわかりません。1枚目のカードと，おしゃべりをヒントに，自分のカード合計を，21を超えないように増やしていくのです。



▲やった21！

決まったら，いよいよ勝負！　勝てば賭けチップが手に入り，負ければその分，相手に奪われます。さあ，あなたは，相手のチップを0にすることができるでしょうか？

CPUの思考ルーチンですが，始めに「目標点」を乱数で決め，そこに到達するようにカードを引いていきます。目標点を超えれば，満足して勝負に来ますが，目標点の近くであれば，諦めてカードを引かなくなることもあります。たったこれだけの乱数ベースの思考ですが，なかなかこれで，手強いですよ。

作り方

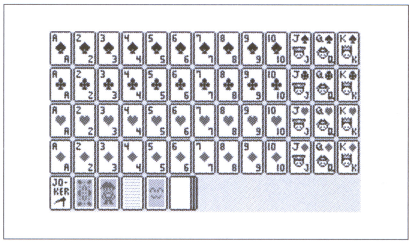
プログラムの準備

①作業場所となるフォルダを作る
エクスプローラなどで，フォルダを作ってください。今回は ¥Piece Program¥blackjack¥ とします。
②元となるファイルをコピーしてくる
（PIECEをインストールしたフォルダ）¥app¥hello から，
・makefile
・hello.c
の二つのファイルをコピーします。
この中のhello.cを書き換え，プログラムを作成します。

データを作成する

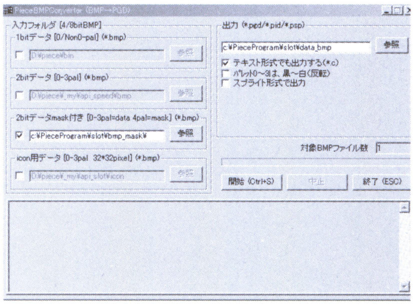
今回のプログラムでは，以下のようなデータが必要になります。

①データの元になるBitmap画像を作成する
・カード画像
[bmp_trump.bmp]



画像詳細
サイズ：画像自体は 208 ×120 ドット
カード1枚は 16 × 24 ドット。
画像ではジョーカーがありますが，今回は使いません。カードの回りは透明色。パレット4番で塗りつぶしてください。

これらのファイルを，以下のように配置してください。
¥PieceProgram¥slot¥bmp_mask¥bmp_trump.bmp
②データをPIECEプログラムで使えるように変換する
（PIECEをインストールしたフォルダ）¥bin¥pBmpCnv.exe を使い，画像を変換します。以下のように指定してください。



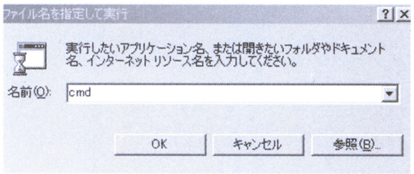
コンパイル

①コンソールを開いてカレントディレクトリを移動する
コンパイルは，コンソールウィンドウを用いて行います。
Windowsのスタートメニューから，「ファイル名を指定して実行」を選び，

・Windows98/Meの場合は「command」
・Windows2000/XPの場合は「cmd」

と打ち込み，実行します。コンソールウィンドウ（DOS窓）が開いたら，

cd ¥PieceProgram¥blackjack
と打ち込み，カレントディレクトリをプログラムフォルダに移動します。



②コマンドをそれぞれ打ち込む
コンパイル（実行ファイルを作る）を行う時は
make
作った実行ファイルをPIECE上に転送して動かすには
run
PIECEにWinlsdで転送できるpexファイルを作るには
make pex
とそれぞれ打ち込みます。

《リスト》トランプゲーム「ブラックジャック」のプログラム・リスト

```
/*
** トランプ：BlackJack
*/

// P/ECE APIを使う為に必要なヘッダ類
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <piece.h>

// 画像データ
#include "data_bmp¥¥bmp_trump.c"

// 画面バッファメモリ( 128 * 88 )
unsigned char vbuff[128*88];

// ゲーム用のデータ ////////////////////////////////////////

// 手札が置かれる場所
const int tblPutField[][2] = {
  { 2, 32 }, { 2, 2 },
};
// ベース文字表示場所
const int tblPutBaseMsg[][2] = {
  { 40, 46 }, // あなた 15
  { 40, 2 },  // ジュン 15
};
// メッセージ表示場所
const int tblPutMsg[] = { 2, 58 };
const int tblWaitCur[] = { 128 / 2 - 5, 88 - 10 };
// セレクタ表示場所
const int tblPutSel[] = { 90, 10 };
// デッキデータ
int deck[52], deckcnt;
// メッセージ表示用
char szMsg[ 1024 ];
int nMsgCnt = 0, nMsgBase = 0;
// カードオープン用
int nCardOpenFlg = 0;
// 掛け金
int nOddNum = 0;

// プレイヤーデータ
typedef struct {
  int    point;   // 持ち点
  int    card[5]; // 場のカード
  int    cardcnt; // 場のカード枚数
} PLAYER;
PLAYER pl[2];

// ゲーム進行ID
enum {
  GM_INIT = 0,       // 初期化
  GM_END,            // メニューに戻る
  GM_PUTMSG,         // メッセージ表示
  GM_PUTMSG_WAIT,    // メッセージウェイト
  GM_START,          // 勝負スタート
  GM_FIRSTGET_CPU,   // 相手の番, 最初の1枚
  GM_FIRSTGET_YOU,   // 自分の番, 最初の1枚
  GM_ODD,            // 賭け金決定
  GM_CPU_TURN,       // 相手の番
  GM_YOU_TURN,       // 自分の番
  GM_OPEN,           // 手札オープン
  GM_NEXT,           // 次の試合へ
};
int nGameId = 0, nNextId;

// P/ECE汎用関数 ////////////////////////////////////////

// 画像パーツ描画関数
void DrawPgdEx( unsigned char *p, int x, int y, int sx, int sy, int sw, int sh, int flg ) {
  PIECE_BMP bmpObj;
  DRAW_OBJECT drawObj;
  memcpy( &(bmpObj.header), p, sizeof( PBMP_FILEHEADER ) );
  bmpObj.buf = p + sizeof( PBMP_FILEHEADER );
  bmpObj.mask = p + sizeof( PBMP_FILEHEADER ) + ( bmpObj.header.h * bmpObj.header.w ) / 4;
  pceLCDSetObject( &drawObj, &bmpObj, x, y, sx, sy, sw, sh, flg );
  pceLCDDrawObject( drawObj );
}

// 塗りつぶし箱
void DrawBoxFill( int nColor, int x1, int y1, int x2, int y2 ) {
  int y;
  unsigned char *pv =pceLCDSetBuffer( INVALIDPTR );
  if ( x2 > 127 ) { x2 = 127; }
  if ( y2 > 87 ) { y2 = 87; }
  if ( x1 < 0 ) { x1 = 0; } if ( x2 < 0 ) { x2 = 0; }
  if ( y1 < 0 ) { y1 = 0; } if ( y2 < 0 ) { y2 = 0; }
  if ( x1 < 128 && y1 < 88 && ( x2 - x1 + 1 ) > 0 ) {
    for ( y = y1; y <= y2; y++ ) {
      memset( &pv[y * 128 + x1], nColor, x2 - x1 + 1 );
    }
  }
}

// 画面バッファ初期化
void DrawClearScreen() {
  unsigned char *pv =pceLCDSetBuffer( INVALIDPTR );
  memset( pv, 0, 88 * 128 );
}

BOOL IsKanji( unsigned char cData ) {
  if ( cData < 0x81 ) { return FALSE; }
  if ( cData < 0xa0 ) { return TRUE; }
  if ( cData < 0xe0 ) { return FALSE; }
  if ( cData < 0xff ) { return TRUE; }
  return FALSE;
}

// ゲームプログラム ////////////////////////////////////////

// デッキからカードを得てカウンタを進める
int GetCard() {
  int n = -1;
  if ( deckcnt < 52 ) {
    n = deck[deckcnt];
    deckcnt++;
  }
  return n;
}

// 全合計を得る
int GetAddNum( int i ) {
  int j, n = 0, z, a = 0;
  for ( j = 0; j < pl[i].cardcnt; j++ ) {
    z = ( pl[i].card[ j ] % 13 ) + 1;
    if ( z > 10 ) { z = 10; }
    if ( z == 1 ) { a++; }
```

```
    n += z;
  }
  if ( n < 12 && a ) { n += 10; }
  return n;
}

// デッキシャッフル
void DeckSort() {
  int j;
  for ( j = 0; j < 52 * 52 * 2; j++ ) {
    int n, m, z;
    n = rand() % 52; m = rand() % 52;
    z = deck[n]; deck[n] = deck[m]; deck[m] = z;
  }
  deckcnt = 0;
}

// ゲーム特有の初期化
void GameInit() {
  int i;
  // デッキの初期化
  for ( i = 0; i < 52; i++ ) { deck[i] = i; }
  // プレイヤーの初期化
  for ( i = 0; i < 2; i++ ) { pl[i].point = 15; }
}

// カード画像の描画
void DrawCard( int num, int x, int y ) {
  if ( num != -1 ) {
    DrawPgdEx( (unsigned char *)BMP_TRUMP, x, y, ( num % 13 ) * 16, ( num / 13 ) * 24, 16,
24, DRW_NOMAL );
  }
}

// メイン画面描画
void DrawMainScreen() {
  int x, y, n, i, j;
  // 画面の初期化
  DrawClearScreen();
  DrawBoxFill( 1, 0, 0, 86, 56 );
  // 持ち点描画
  for ( i = 0; i < 2; i++ ) {
    char sz[][8] = { "あなた", "ジュン" };
    pceFontSetPos( tblPutBaseMsg[i][0], tblPutBaseMsg[i][1] );
    pceFontPrintf( "%s %02d", sz[i], pl[i].point );
  }
  // 手札の描画
  for ( i = 0; i < 2; i++ ) {
    for ( j = 0; j < pl[i].cardcnt; j++ ) {
      if ( i == 0 || j == 0 || ( i == 1 && nCardOpenFlg ) ) {
        DrawCard( pl[i].card[ j ], tblPutField[i][0] + j * 16, tblPutField[i][1] );
      } else {
        DrawCard( 53, tblPutField[i][0] + j * 16, tblPutField[i][1] );
      }
    }
  }
  // メッセージの描画
  x = tblPutMsg[0]; y = tblPutMsg[1];
  for ( i = nMsgBase; i < nMsgCnt; i++ ) {
    n = 1;
    if ( IsKanji( szMsg[ i ] ) ) { n = 2; }
    if ( x >= 128 - n * 5 || szMsg[ i ] == '¥r' ) {
      x = tblPutMsg[0] + 5; y += 10;
      if ( szMsg[ i ] == '¥r' ) { continue; }
    }
    pceFontSetPos( x, y );
    if ( n == 1 ) {
      pceFontPrintf( "%c", szMsg[ i ] );
    } else {
      pceFontPrintf( "%c%c", szMsg[ i ], szMsg[ i + 1 ] );
      i++;
    }
    x += n * 5;
  }
}

// ゲームプロシージャ
void GameProc( int cnt, long lKey ) {
  int i = 0;
  switch( nGameId ) {
    case GM_INIT:
    { // ゲームの初期化
      strcpy( szMsg,
        "ジュン「さあ、ブラック¥rジャックを始めましょう」¥n"
        "ジュン「チップは15枚。¥r一度に賭けられるのは5枚¥rまでです。¥n"
        "ジュン「では、ぼくから¥r引きます。"
      );
      nGameId = GM_PUTMSG; nNextId = GM_START;
      break;
    }
    case GM_END:
    { // ゲームの終了
      pceAppReqExit( 0 );
      break;
    }
    case GM_PUTMSG:
    { // メッセージ表示
      if ( IsKanji( szMsg[ nMsgCnt ] ) ) { nMsgCnt++; }
      nMsgCnt++;
      if ( szMsg[ nMsgCnt ] == '¥n' ||  szMsg[ nMsgCnt ] == '¥0' ) {
        nGameId = GM_PUTMSG_WAIT;
      }
      // 画面描画
      DrawMainScreen();
      break;
    }
    case GM_PUTMSG_WAIT:
    { // メッセージウェイト
      // 画面描画
      DrawMainScreen();
      if ( lKey & TRG_A ) {
        if ( szMsg[ nMsgCnt ] == '¥n' ) {
          nGameId = GM_PUTMSG; nMsgCnt++; nMsgBase = nMsgCnt;
        } else if ( szMsg[ nMsgCnt ] == '¥0' ) {
          nGameId = nNextId; nMsgCnt = 0; nMsgBase = 0;
        }
      }
      if ( cnt % 30 < 20 ) {
        pceFontSetPos( tblWaitCur[0], tblWaitCur[1] );
        pceFontPrintf( "▼" );
      }
      break;
    }
    case GM_START:
    { // ゲームスタート
      nCardOpenFlg = 0;
      DeckSort();
      for ( i = 0; i < 2; i++ ) { pl[i].cardcnt = 0; }
      nGameId = GM_FIRSTGET_CPU;
```

```
    break;
  }
  case GM_FIRSTGET_CPU:
  { // 相手の番，最初の1枚
    i = 1;
    pl[i].card[ pl[i].cardcnt ] = GetCard();
    pl[i].cardcnt++;
    strcpy( szMsg, ”ジュン「ボス。１枚目を¥r引いて下さい。” );
    nGameId = GM_PUTMSG;
    nNextId = GM_FIRSTGET_YOU;
    // 画面描画
    DrawMainScreen();
    break;
  }
  case GM_FIRSTGET_YOU:
  { // 自分の番，最初の1枚
    i = 0;
    pl[i].card[ pl[i].cardcnt ] = GetCard();
    pl[i].cardcnt++;
    strcpy( szMsg, ”ジュン「さあボス，賭けて下さい。何枚賭けますか？” );
    nGameId = GM_PUTMSG;
    nNextId = GM_ODD;
    // 画面描画
    DrawMainScreen();
    break;
  }
  case GM_ODD:
  { // 賭け金選択
    static int nCur = 0, nCnt = 0;
    char sz[][8] = { ”１枚”, ”２枚”, ”３枚”, ”４枚”, ”５枚”, ”やめる”, };
    // セレクト
    if ( lKey & TRG_UP ) { nCur--; nCnt = 0; }
    if ( lKey & TRG_DN ) { nCur++; nCnt = 0; }
    if ( nCur < 0 ) { nCur = 5; }
    if ( nCur > 5 ) { nCur = 0; }
    if ( lKey & TRG_A ) {
      nMsgCnt = 0;
      if ( nCur != 5 ) {
        sprintf( szMsg,
          ”ボス「%s¥n”
          ”ジュン「%sですね。¥rでは，ぼくから引きます。”
          , sz[nCur], sz[nCur] );
        nNextId = GM_CPU_TURN;
        nOddNum = nCur + 1;
      } else {
        sprintf( szMsg,
          ”ボス「%s¥n”
          ”ジュン「おや？　もう参ったですか？　いやー，ボスは弱いなあ。¥n”
          ”ジュン「では，メニューに戻ります。”
          , sz[nCur] );
        nNextId = GM_END;
      }
      nGameId = GM_PUTMSG;
      nCur = 0; nCnt = 0;
    } else {
      strcpy( szMsg, ”ボス「？” ); nMsgCnt = strlen( szMsg );
    }
    // 画面描画
    DrawMainScreen();
    for ( i = 0; i < 6; i++ ) {
      pceFontSetPos( tblPutSel[0] + 5, tblPutSel[1] + i * 10 );
      pceFontPrintf( sz[i] );
    }
    if ( nCnt % 30 < 20 ) {
      pceFontSetPos( tblPutSel[0], tblPutSel[1] + nCur * 10 );
      pceFontPrintf( ”>” );
    }
    nCnt++;
    break;
  }
  case GM_CPU_TURN:
  { // 相手の番
    static int nMode = 0, nTarget;
    if ( nMode == 0 ) {
      // CPUの目標値を決める
      nTarget = 21 - ( rand() % 5 );
      nMode = 1;
    } else if ( nMode == 1 ) {
      // 1枚取る
      i = 1;
      pl[i].card[ pl[i].cardcnt ] = GetCard();
      pl[i].cardcnt++;
      nMode = 2;
    } else if ( nMode == 2 ) {
      // 思考
      int n, r;
      n = GetAddNum( 1 ); r = ( rand() % 4 ) + 1;
      // オーバー判定
      if ( n > 21 ) {
        // オーバー
        nGameId = GM_OPEN; nMode = 0;
      } else {
        // CPU思考
        if ( n >= nTarget ) {
          // OK.これ以上取らない
          r = rand() % 3;
          if ( r == 0 ) {
            strcpy( szMsg, ”ジュン「やった。¥rばっちりだぜっ！” );
          } else if ( r == 1 ) {
            strcpy( szMsg, ”ジュン「こりゃあ，¥r勝ったな。” );
          } else if ( r == 2 ) {
            strcpy( szMsg, ”ジュン「わあ！¥rいいのが来たなあ。” );
          }
          nNextId = GM_YOU_TURN; nGameId = GM_PUTMSG;
          nMode = 0;
        } else if ( n < 12 && pl[1].cardcnt < 5 ) {
          // 取る
          strcpy( szMsg, ”ジュン「えい，もう１枚！” );
          nNextId = nGameId; nGameId = GM_PUTMSG;
          nMode = 1;
        } else if ( n >= nTarget - r || pl[1].cardcnt == 5 ) {
          // 悩んで取らない
          r = rand() % 3;
          if ( r ) {
            strcpy( szMsg, ”ジュン「うわっ！¥rまいっちゃったな。¥n” );
          } else {
            strcpy( szMsg, ”ジュン「うーん，¥rどうしようかなあ。¥n” );
          }
          strcat( szMsg, ”ジュン「これでいいや。” );
          nNextId = GM_YOU_TURN; nGameId = GM_PUTMSG;
          nMode = 0;
        } else {
          // 悩んで取る
          r = rand() % 3;
          if ( r ) {
            strcpy( szMsg, ”ジュン「うわっ！¥rまいっちゃったな。¥n” );
          } else {
            strcpy( szMsg, ”ジュン「うーん，¥rどうしようかなあ。¥n” );
```

```
          }
          strcat( szMsg, "ジュン「えい，もう１枚！" );
          nNextId = nGameId; nGameId = GM_PUTMSG;
          nMode = 1;
        }
      }
    }
    // 画面描画
    DrawMainScreen();
    break;
  }
  case GM_YOU_TURN:
  { // 自分の番
    static int nMode = 0, nCur = 0, nCnt = 0;
    if ( nMode == 0 ) {
      // 1枚取る
      strcpy( szMsg, "ジュン「では，ボスが¥r引く番です。" );
      nNextId = nGameId; nGameId = GM_PUTMSG;
      nMode = 1;
    } else if ( nMode == 1 ) {
      i = 0;
      pl[i].card[ pl[i].cardcnt ] = GetCard();
      pl[i].cardcnt++;
      nCur = 0; nCnt = 0;
      if ( GetAddNum( 0 ) > 21 ) {
        // オーバー
        nGameId = GM_OPEN; nMode = 0;
      } else if ( pl[i].cardcnt == 5 ) {
        // 枚数終わり
        sprintf( szMsg, "ジュン「さあ，勝負です！" );
        nNextId = GM_OPEN; nGameId = GM_PUTMSG;
        nMode = 0;
      } else {
        nCur = 0; nCnt = 0;
        strcpy( szMsg, "ジュン「もう１枚，¥r引きますか？" );
        nNextId = nGameId; nGameId = GM_PUTMSG;
        nMode = 2;
      }
    } else {
      char sz[][8] = { "はい", "いいえ", };
      // セレクト
      if ( lKey & TRG_UP ) { nCur--; nCnt = 0; }
      if ( lKey & TRG_DN ) { nCur++; nCnt = 0; }
      if ( nCur < 0 ) { nCur = 1; }
      if ( nCur > 1 ) { nCur = 0; }
      if ( lKey & TRG_A ) {
        nMsgCnt = 0;
        if ( nCur == 0 ) {
          sprintf( szMsg,
            "ボス「%s"
            , sz[nCur] );
          nNextId = nGameId;
          nMode = 1;
        } else {
          sprintf( szMsg,
            "ボス「%s¥n"
            "ジュン「さあ，勝負です！"
            , sz[nCur] );
          nNextId = GM_OPEN;
          nMode = 0;
        }
        nGameId = GM_PUTMSG;
      } else {
        strcpy( szMsg, "ボス「？" ); nMsgCnt = strlen( szMsg );
```

```
      }
      // 画面描画
      DrawMainScreen();
      for ( i = 0; i < 2; i++ ) {
        pceFontSetPos( tblPutSel[0] + 5, tblPutSel[1] + i * 10 );
        pceFontPrintf( sz[i] );
      }
      if ( nCnt % 30 < 20 ) {
        pceFontSetPos( tblPutSel[0], tblPutSel[1] + nCur * 10 );
        pceFontPrintf( ">" );
      }
      nCnt++;
    }
    break;
  }
  case GM_OPEN:
  { // 手札オープン
    int r;
    r = rand() % 2;
    nCardOpenFlg = 1;
    if ( GetAddNum( 1 ) > 21 ) {
      strcpy( szMsg, "ジュン「まいっちゃった¥rな。オーバーです。" );
    } else if ( GetAddNum( 0 ) > 21 ) {
      strcpy( szMsg, "ジュン「おや，オーバー¥rですか？　残念でしたね。" );
    } else {
      sprintf( szMsg, "ジュン「ぼくは，%dです。¥n", GetAddNum( 1 ) );
      if ( GetAddNum( 0 ) == GetAddNum( 1 ) ) {
        sprintf( szMsg, "%sジュン「おや？¥r引き分けですね。", szMsg );
      } else if ( GetAddNum( 0 ) > GetAddNum( 1 ) ) {
        if ( r == 0 ) {
          sprintf( szMsg, "%sジュン「%dですか？¥rまいっちゃったな。", szMsg, GetAddNum( 0 )
);
        } else if ( r == 1 ) {
          sprintf( szMsg, "%sジュン「いやあ，ボスは¥r強いなあ。", szMsg );
        }
      } else if ( GetAddNum( 0 ) < GetAddNum( 1 ) ) {
        if ( r == 0 ) {
          sprintf( szMsg, "%sジュン「ボスは%dですか？¥rわるいなあ。あはは。", szMsg,
GetAddNum( 0 ) );
        } else if ( r == 1 ) {
          sprintf( szMsg, "%sジュン「どうやら，ぼくの勝ちみたいですね。", szMsg );
        }
      }
    }
    nNextId = GM_NEXT; nGameId = GM_PUTMSG;
    break;
  }
  case GM_NEXT:
  { // 次の試合へ
    int n0, n1;
    n0 = GetAddNum( 0 ); n1 = GetAddNum( 1 );
    if ( n0 > 21 ) { n0 = 0; }
    if ( n1 > 21 ) { n1 = 0; }
    if ( n0 > n1 ) {
      pl[0].point += nOddNum;
      pl[1].point -= nOddNum;
    } else if ( n0 < n1 ) {
      pl[0].point -= nOddNum;
      pl[1].point += nOddNum;
    }
    if ( pl[0].point < 0 ) { pl[0].point = 0; }
    if ( pl[1].point < 0 ) { pl[1].point = 0; }
    if ( pl[0].point <= 0 ) {
```

```
      strcpy( szMsg,
        ”ジュン「おや？¥rボスはもう・・¥n”
        ”ジュン「チップが無くなってしまったみたいですね。¥n”
        ”ジュン「この勝負，¥rぼくの勝ち！¥n”
        ”ジュン「さあ，メニューに戻りましょう！”
      );
      nNextId = GM_END;
    } else if ( pl[1].point <= 0 ) {
      strcpy( szMsg,
        ”ジュン「あれれ？¥n”
        ”ジュン「もうチップが無くなっちゃったいっ。¥n”
        ”ジュン「さあ，メニューに戻りましょう！¥n”
        ”ジュン「ところでボス。ぼくは思うんですがね・・。¥n”
      );
      strcat( szMsg,
        ”ジュン「こんなサンプルで¥rいいんですかね？”
      );
      nNextId = GM_END;
    } else {
      strcpy( szMsg, ”ジュン「では，勝負を¥r続けましょう。” );
      nNextId = GM_START;
    }
    nGameId = GM_PUTMSG;
    break;
  }
 }
 // 画面バッファの内容を液晶に転送
 pceLCDTrans();
}

// P/ECEシステム関数 ////////////////////////////////////////

// プログラム起動時に１度だけ実行される関数です
void pceAppInit( void ) {
  PCETIME strTime;
  // 画面周りの初期化を行う
  pceLCDDispStop();
  pceLCDSetBuffer( vbuff );
  memset( vbuff, 0, 128*88 );
  // pceAppProc()が呼び出される間隔を指定(ミリ秒)
  pceAppSetProcPeriod( 22 );
  // 乱数初期化
  pceTimeGet( &strTime ); srand( strTime.s100 );
  // フォントの初期化
  pceFontSetType( 0 + 128 ); pceFontSetBkColor( -1 ); pceFontSetTxColor( 3 );
  // ゲーム特有の初期化
  GameInit();
  // 画面の更新を開始する
  pceLCDDispStart();
}

// 一定時間( pceAppSetProcPeriod()で指定した間隔) 毎に呼び出される
// プログラムのメインになる関数です
void pceAppProc( int cnt ) {
  GameProc( cnt, pcePadGet() );
}

// プログラム終了時に１度だけ実行される関数です
void pceAppExit( void ) {

}
```

PIECE関連，大々的に募集中！

「P/ECE HANDBOOK Vol.2」がついに発売されましたが，ベーマガ本誌，そして夢のVol.3に向けて，PIECEの投稿を大募集中です。発表する場はこれからも数多く用意されるので，ぜひぜひ，PIECEでプログラミング経験値を上げてしまいましょう！

・PIECEモニター

PIECEに触ってみたい，遊んでみたい人！　ならば是非ぜひ，体験してみてください。PIECE本体と，電波新聞社発売の単行本「P/ECE HANDBOOK」をセットでお貸し致します。個人でも，学校のクラブでもOKですよ！
※当選された方には，使ってみての感想やレポートを提出していただきます。

・PIECE関連Webサイト

インターネットでPIECE関連のWebサイトを運営している人，はたまた，お気に入りの関連Webサイトがある人。ぜひ，本連載で紹介させてください！　そのサイトのURLと，オススメポイントを明記してくださいね。

・画像データ，音データ，ネタ等

PIECEで使えるデータや，「こんなことを連載でやって欲しい」というネタまで広く募集しちゃいます。あなたの絵やアイデアを，この連載で形にします！

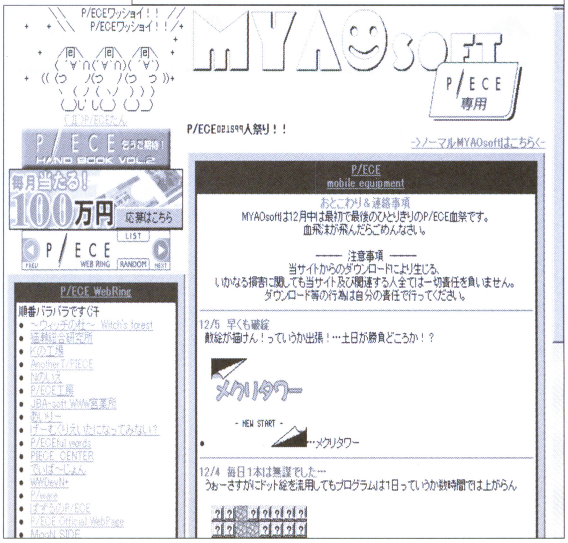
・オリジナルプログラム

PIECEで動くオリジナルプログラムももちろん募集！　こちらは，本誌投稿プログラムコーナーを参照くださいね。PIECEプログラムで原稿料を貰ってしまおう！

夢が広がる応募はこちらへ！
〒141-8715
東京都品川区東五反田1-11-15
電波新聞社「マイコンBASICマガジン編集部」PIECE係

今月のPIECE厳選Webサイト

MYAOsoft
(http://www.myaosoft.com/)
2002年12月現在，PIECE一周年を記念して，PIECE祭りを開催している活気あるWebサイトです。一日一本，ソフトを作ることを目標に頑張っています。多くのソフトをダウンロードしたい人はこちらから！



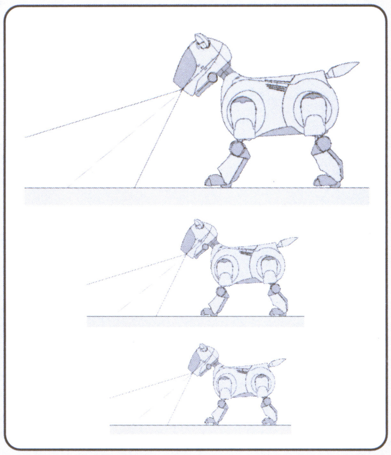
21世紀の遊びはコレ！

ロボ道場

「OPEN-R SDK」プログラミング②

AIBOで画像処理に挑戦

文／松原拓也



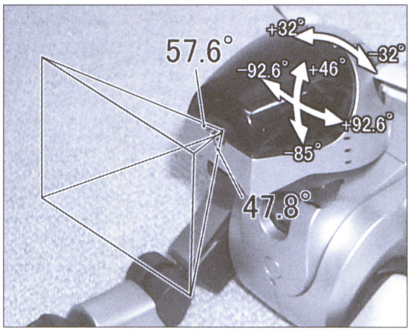
今回は「OPEN-R SDK」を使って，AIBOの頭部に搭載されているカメラを利用して，画像処理を行ったプログラミングをしてみましょう。

OPEN-R SDKの動作環境

《PC環境》
・OS : Windows 2000 Professional/ XP
・"メモリースティック"スロットをPCで使うためのシステム
・PC側にIEEE802.11b準拠のWireless LANカード（または LANに接続された IEEE802.11b準拠のアクセスポイント）
《AIBO》
・AIBO (ERS-210/220/210A/220A) 本体
・AIBOプログラミングメモリースティック(ERA-MS008 または ERA-MS016)
・AIBOワイヤレスLANカードERA-201D1
※“Remote Processing OPEN-R”を利用することで，AIBOがなくてもOPEN-Rオブジェクトのデバッグやテスト実行ができます。
(注:AIBOがなくても画像キャプチャや関節が動かせると思われるので)ただし，AIBO との通信がなければAIBO の機能(センサー入力，画像入力，音声入力，関節モータ制御，音声出力，LED点灯など)が使えないので，ロボット制御として有意なことはできません。
「OPEN-R SDK」WEBサイト
http://www.jp.aibo.com/openr/

AIBOで写真撮影

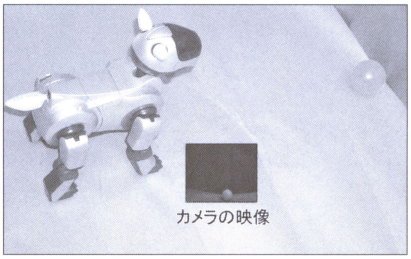
では，「ImageCapture」というサンプルプログラムで，写真を撮ってみましょう（「OPEN-R SDK」WEBサイトにて公開中です）。ファイルをダウンロードしたら，解凍・ビルドを行って，AIBOプログラミング･メモリースティックに書き込んでください（開発方法は2002年1月号を参照してください）。プログラムの実行後，AIBOの背中センサを押すと，カメラ（**写真1**）で撮影され，その映像がBMPファイルとしてメモリースティックに書き込まれます（**写真2**）。BMPファイルはFTPでログインするか，メモリースティックリードライターで取り出せます（**写真3**）。



《写真1》カメラの画角と頭部の可動範囲

```
接続(C) 編集(E) ターミナル(T) ヘルプ(H)
sol_bye2tls_xld 3
commandID 115 numAgentCommands 2 useSyncKey 1
a_sit#sit_so0_byebye2r 1
sol_bye2rtls_xld 2
commandID 116 numAgentCommands 2 useSyncKey 1
a_sleep#sleep_so4_aisatsu1 5
sol_aisatsu1pxs_xld 1
Node 1 : (1->4)
Node 2 : (2->3)(2->4)(2->5)
Node 3 : (3->4)(3->2)
Node 4 : (4->3)(4->2)
Node 5 : (5->2)
Debug settings for TCP endpoint in IPStack
set by registry key TCPEP_DEBUG
registry key NOT set
debug output for TCP endpoint in IPStack is disabled
AGENT RESULT : agent 0 index 0 status 1 endPos 4
AGENT RESULT : agent 1 index 6 status 1 endPos 2
AGENT RESULT : agent 1 index -1 status 1 endPos 2
writing /MS/OPEN-R/MW/DATA/P/RGBH0000.BMP ...
writing /MS/OPEN-R/MW/DATA/P/LAYH0000.RAW ...
writing /MS/OPEN-R/MW/DATA/P/LAYM0000.RAW ...
writing /MS/OPEN-R/MW/DATA/P/LAYL0000.RAW ...
done.
```

《写真2》サンプル「ImageCapture」を実行した時のtelnetの画面。カメラの映像はBMPで保存される



《写真3》「ImageCapture」実行時の様子

画像認識に挑戦

AIBOのカメラの正式名は「イメージセンサ」。センサという名のとおり，「OPEN-R」ではカメラで物事を判断させることができます。

たとえば，サンプルプログラムの「BallTrackingHead」は，ピンクボールの位置に合わせてAIBOの頭を追跡（トラッキング）することができます（「OPEN-R SDK」WEBサイトで公開中です）。

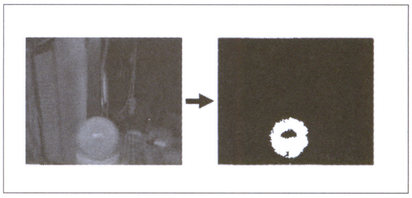
この処理では，AIBOに搭載されている「色検出エンジン」というハードウェアが，ピンクボールのピンク色を検出しています（**写真4**）。

ボール検出の流れ

少々ややこしいのですが，ボールを検出するまでの「BallTrackingHead」の処理手順を追ってみましょう。

まず，オブジェクトで最初に実行される「DoInit」関数で，プリミティブIDの取得（OpenPrimitive関数）と，色検出テーブル（SetCdtVectorDataOfPinkBall関数）の設定を行います。

プリミティブIDとは，デバイス機器の



《写真4》カメラ映像から特定色を抜き出す

ハンドルのようなものです。IDの取得には「OpenPrimitive」関数を使います。カメラ（イメージセンサ）のロケータ名は“"PRM:/r1/c1/c2/c3/i1-FbkImageSensor:F1”です。

色検出テーブル

先ほど紹介した「色検出エンジン」に，検出する色を格納させているのが「色検出テーブル（略してCDT）」です。CDTには最大8色（チャンネル）まで登録できます。

新しくCDTに登録するには「CDTベクターデータ」というデータを作らないといけません。まず「NewCdtVectorData」関数でCDTベクターデータを準備して，「SetNumData」でチャンネル数の設定，次に「GetInfo」で先頭ポインタの取得，「Init」で一旦初期化を行い，「Set」命令で色の範囲を設定します（**リスト1**）。Setの引数は以下のとおりです。

Set(輝度Yのセグメント0〜31,
Cr最大値0〜255, Cr最小値0〜255,
Cb最大値0〜255, Cb最小値0〜255)

カメラの映像は，RGBではなく，YCbCrという形式で扱います（Y＝輝度，Cr＝赤の色差(輝度―赤)，Cb＝青の色差(輝度―青)）。YCbCrはRGBと違って「緑」の情報が抜けていますが，以下の計算式で求めることができます。

R ＝ 255.0 ＊（Y＋ Cr);
G＝255.0＊(Y－0.51＊Cr－0.19＊Cb);
B ＝ 255.0＊（Y＋ Cb);
(この式はYの範囲を0〜＋1.0，CrとCbを-1.0〜＋1.0とした場合のものです)

輝度は32段階に分けて，その輝度ごとに範囲を設定できます。たとえば，ピンク色を検出する場合，以下のようになります。

cdt－>Set(0, 230, 150, 190, 120);
..中略...
cdt－>Set(31, 230, 160, 190, 120);

このCb・Crの値を変えると，ピンク以外の色に反応させることができます（**写真5**）。たとえば，以下のように修正することで，蛍光の緑色に反応します。

cdt－>Set(0, 110, 50, 110, 50);
..中略...
cdt－>Set(31, 110, 50, 110, 50);

次に「SetCdtVectorData」関数で色検出テーブルに登録します。最後に「DeleteCdtVectorData」関数で，必要となくなったCDTベクターデータを開放します。

画像データの取得

カメラの画像データは，オブジェクト間通信を使って，「OVirtualRobotComm.FbkImageSensor」というサブジェクトから受け取ります（オブジェクト間通信については後で説明します)。この時，オブザーバとして設定している「Notify」が呼び出されます。「Notify」の最後には，次の画像データをもらうため，「AssertReady」を呼び出します。

画像データは「event.Data(0)」で取り出します。データの形式は，「OFbkImageVectorData」という構造体のポインタです。

受け取った画像データ(レイヤー)には4種類あります。

《写真5》YCrCbのカラーテーブル（モノクロページですが）

・「レイヤーH」高解像度YCbCr画像。176×144pixel。
・「レイヤーM」中解像度YCbCr画像。88×72pixel。
・「レイヤーL」低解像度YCbCr画像。44×36pixel。
・「レイヤーC」色検出エンジンの処理結果(CDT画像)。88×72。

レイヤーH/M/Lは大きさに分けて使う画像データですが，今回は使いません。レイヤーCは，色検出ずみの画像なので，ピンク色だけに反応した画像です。レイヤーCの画像データが取り出すためには，以下のように「OFbkImage」クラスを作成します。

角度の算出

レイヤーCの値を元に，カメラに映っている目標物の角度を割り出します。OFbkImageクラスは「クラス

オブジェクト間通信

「オブジェクト間通信」とは，オブジェクト（プログラムのモジュール）が別のオブジェクトに対してデータを送受信することです。そして，データの送信元を「サブジェクト」，データの受信先を「オブザーバ」といいます。オブジェクト間通信の処理の流れは以下のようになっています。

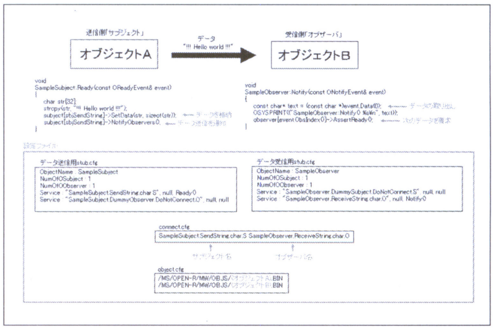
①サブジェクト側でReady関数が実行される。SetDataで送りたいデータを格納する。

②送信完了の合図として，サブジェクトがNotifyObserver関数を実行する。

③オブザーバ側でNotify関数が実行される。受け取ったデータはevent.Dataで取り出すことができる。

④受信完了の合図として，オブザーバがAssertReady関数を実行。処理①に戻る。

あと，オブジェクト間通信を行うには，“STUB.CFG”という設定ファイルに「サブジェクト数とオブザーバ数」「オブジェクト間通信で使用するサービス」を記述して，さらに“CONNECT.CFG”というファイルにサブジェクトとオブザーバの接続情報を記述する必要があります。



▲オブジェクト間通信のサンプル

名.Pixel(x, y)」とすることで，その座標のカラーを知ることができます。そこで，「CentroidAndDeltaAngle」関数では検出した色のX座標とY座標の平均を求めます。この座標がボールの中心になります。そして，ボールの座標とカメラの画角から，ボールの角度を求めます。カメラの画角は横方向に57.6°，縦方向に47.8°です。

ボールの角度
＝カメラの画角×座標÷画像の幅

頭部の可動範囲は首パンが−92.6〜＋92.6°，首チルトが−85〜＋46°，首ロールが−32〜＋32°です。プログラムでは，パン方向・チルト方向ともに1回に最大5°だけ動くようになっています（TrackBall関数）。

ボール追跡プログラムを自作

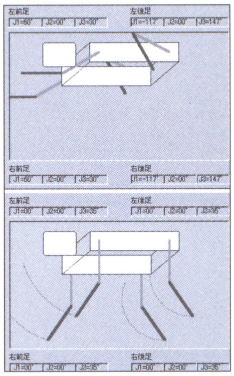
この「BallTrackingHead」を改造してボール追跡プログラムを自作してみました。

・色検出で，目標の中心に首を向け続ける（トラッキング）。
・首の角度がまっすぐになるように，体を方向転換させる。
・首がまっすぐを向いているなら前進する。
・（以下，繰り返し）

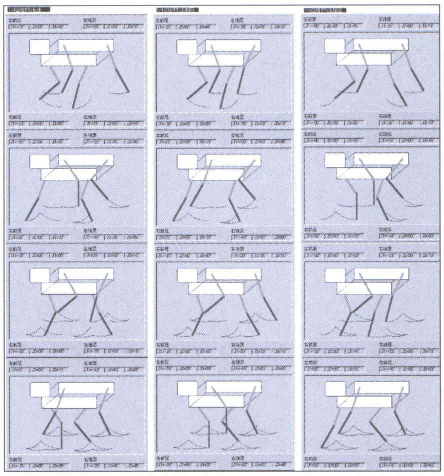
先月号（2003年1月号）では，直進しかできませんでしたが，**写真6**や**写真7**のように，立ち上がったり，旋回できるようにしました。

画像処理はOPEN-R SDKの醍醐味！

なお，「OPEN-R SDK」のWEBサイトでは，同様にボールを追いかけるプログラムとして，「SoccerLion」というサンプルが公開されています。こちらでは，ボールが近づいたら，ボールをキックするという機能も盛り込まれています。ぜひ参考にしてみてください。



《写真6》置き上がり動作時の角度



《写真7》前進，右旋回，左旋回時の足の動き

《写真8》ピンクボールの追跡に挑戦

《リスト》サンプルの「BallTrackingHead」より“BallTrackingHead_Image.cc”

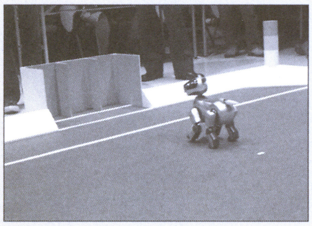
```
//
// Copyright 2002 Sony Corporation
//
// Permission to use, copy, modify, and redistribute this software for
// non-commercial use is hereby granted.
//
// This software is provided "as is" without warranty of any kind,
// either expressed or implied, including but not limited to the
// implied warranties of fitness for a particular purpose.
//

#include <OPENR/OPENRAPI.h>
#include <OPENR/OSyslog.h>
#include "BallTrackingHead.h"

void
BallTrackingHead::NotifyImage(const ONotifyEvent& event)
{
    static int found = 0;
    static int lost = 0;

    int    xc, yc;                // centroid
    double d_pan, d_tilt;         // delta angle

    if (ballTrackingHeadState == BTHS_IDLE) return; // do nothing

    OFbkImageVectorData* imageVec = (OFbkImageVectorData*)event.Data(0);

    OFbkImageInfo* info = imageVec->GetInfo(ofbkimageLAYER_C);
    byte*          data = imageVec->GetData(ofbkimageLAYER_C);
    OFbkImage cdtImage(info, data, ofbkimageBAND_CDT);  /*レイヤーCの画像を
取得*/

    if (ballTrackingHeadState == BTHS_SEARCHING_BALL) {
        if (cdtImage.ColorFrequency(BALL_CDT_CHAN) >= BALL_THRESHOLD) {
            /* ピンクの色頻度を判定 */
            found++;
        } else {
            found = 0;
            SearchBall();
        }

        if (found == FOUND_THRESHOLD) {
            OSYSPRINT(("### BALL FOUND ### ¥n"));
            PlaySound(BTHCMD_PLAY_FOUND_SOUND);
            found = 0;
            CentroidAndDeltaAngle(cdtImage, &xc, &yc, &d_pan, &d_tilt); /*角
度の算出*/
            TrackBall(info->frameNumber, d_pan, d_tilt); /*首の移動*/
        }

    } else if (ballTrackingHeadState == BTHS_TRACKING_BALL) {/*トラッキング
中*/

        if (cdtImage.ColorFrequency(BALL_CDT_CHAN) >= BALL_THRESHOLD) {
            lost = 0;
            CentroidAndDeltaAngle(cdtImage, &xc, &yc, &d_pan, &d_tilt); /*角
度の算出*/
            TrackBall(info->frameNumber, d_pan, d_tilt); /*首の移動*/
        } else {
            lost++;
        }

        if (lost == LOST_THRESHOLD) {
            OSYSPRINT(("### BALL LOST ### ¥n"));
            PlaySound(BTHCMD_PLAY_LOST_SOUND);
            SearchBall();
            lost = 0;
        }
    }

    observer[event.ObsIndex()]->AssertReady(); /*次のデータ受信OKを通知*/
}

/*色検出テーブルの設定*/
void
BallTrackingHead::SetCdtVectorDataOfPinkBall()
{
    OSYSDEBUG(("BallTrackingHead::SetCdtVectorDataOfPinkBall()¥n"));

    OStatus result;
    MemoryRegionID  cdtVecID;
    OCdtVectorData* cdtVec;
    OCdtInfo*       cdt;

    result = OPENR::NewCdtVectorData(&cdtVecID, &cdtVec);
    if (result != oSUCCESS) {
        OSYSLOG1((osyslogERROR, "%s : %s %d",
                  "ImageObserver::SetCdtVectorDataOfPinkBall()",
                  "OPENR::NewCdtVectorData() FAILED", result));
```

```
        return;
    }

    cdtVec->SetNumData(1);          /*使用するテーブル数*/

    cdt = cdtVec->GetInfo(0);
    cdt->Init(fbkID, BALL_CDT_CHAN); /*初期化*/

    //
    // cdt->Set(Y_segment, Cr_max,  Cr_min, Cb_max, Cb_min)
    //
    cdt->Set( 0, 230, 150, 190, 120);          /*色検出範囲の設定*/
    cdt->Set( 1, 230, 150, 190, 120);
    cdt->Set( 2, 230, 150, 190, 120);
    cdt->Set( 3, 230, 150, 190, 120);
    cdt->Set( 4, 230, 150, 190, 120);
    cdt->Set( 5, 230, 150, 190, 120);
    cdt->Set( 6, 230, 150, 190, 120);
    cdt->Set( 7, 230, 150, 190, 120);
    cdt->Set( 8, 230, 150, 190, 120);
    cdt->Set( 9, 230, 150, 190, 120);
    cdt->Set(10, 230, 150, 190, 120);
    cdt->Set(11, 230, 150, 190, 120);
    cdt->Set(12, 230, 150, 190, 120);
    cdt->Set(13, 230, 150, 190, 120);
    cdt->Set(14, 230, 150, 190, 120);
    cdt->Set(15, 230, 150, 190, 120);
    cdt->Set(16, 230, 150, 190, 120);
    cdt->Set(17, 230, 150, 190, 120);
    cdt->Set(18, 230, 150, 190, 120);
    cdt->Set(19, 230, 150, 190, 120);
    cdt->Set(20, 230, 160, 190, 120);
    cdt->Set(21, 230, 160, 190, 120);
    cdt->Set(22, 230, 160, 190, 120);
    cdt->Set(23, 230, 160, 190, 120);
    cdt->Set(24, 230, 160, 190, 120);
    cdt->Set(25, 230, 160, 190, 120);
    cdt->Set(26, 230, 160, 190, 120);
    cdt->Set(27, 230, 160, 190, 120);
    cdt->Set(28, 230, 160, 190, 120);
    cdt->Set(29, 230, 160, 190, 120);
    cdt->Set(30, 230, 160, 190, 120);
    cdt->Set(31, 230, 160, 190, 120);

    result = OPENR::SetCdtVectorData(cdtVecID); /*CDTベクターデータを設定*/
    if (result != oSUCCESS) {
        OSYSLOG1((osyslogERROR, "%s : %s %d",
                  "ImageObserver::SetCdtVectorDataOfPinkBall()",
                  "OPENR::SetCdtVectorData() FAILED", result));
    }

    result = OPENR::DeleteCdtVectorData(cdtVecID); /*CDTベクターデータを開放
*/
    if (result != oSUCCESS) {
        OSYSLOG1((osyslogERROR, "%s : %s %d",
                  "ImageObserver::SetCdtVectorDataOfPinkBall()",
                  "OPENR::DeleteCdtVectorData() FAILED", result));
    }
}

/*ピンクボールの存在する角度を算出*/
bool
BallTrackingHead::CentroidAndDeltaAngle(const OFbkImage& cdtImage,
                                        int* xcentroid, int* ycentroid,
                                        double* delta_pan, double*
delta_tilt)
{
    int w     = cdtImage.Width();
    int h     = cdtImage.Height();
    byte* ptr = cdtImage.Pointer();

    int xsum    = 0;
    int ysum    = 0;
    int npixels = 0;

    // Last line is not used because of TagInfo.
    for (int y = 0; y < h-1; y++) {
        for (int x = 0; x < w; x++) {
            if (*ptr++ & BALL_CDT_CHAN) {
                xsum += x;         /*ピンクボールの座標を累積*/
                ysum += y;
                npixels++;
            }
        }
    }

    if (npixels != 0) {
        *xcentroid  = xsum / npixels;         /*ピンクボールの平均座標を求め
る*/
        *ycentroid  = ysum / npixels;
        *delta_pan  = -1.0 * FIELD_VIEW_H * (*xcentroid - w/2.0) / w; /*角度
の算出*/
        *delta_tilt = -1.0 * FIELD_VIEW_V * (*ycentroid - h/2.0) / h;
        return true;
    } else {
        *xcentroid  = 0;
        *ycentroid  = 0;
        *delta_pan  = 0.0;
        *delta_tilt = 0.0;
        return false;
    }
}
```

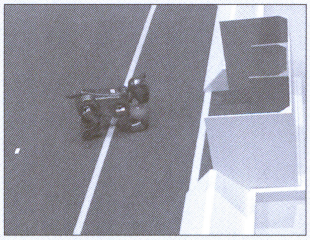
「OPEN-R TECHNO FORUM 2002 in Japan」

2002年12月1日，日本科学未来館にて「OPEN-R TECHNO FORUM 2002 in Japan」が開催されました。同イベントでは，「OPEN-R SDK」で自作プログラムを組み込んだAIBOを使った競技会や，「OPEN-R SDK」に関する技術発表が行われました。

行われた競技は，AIBOによる「PK戦」と「レース競技」の2種目。

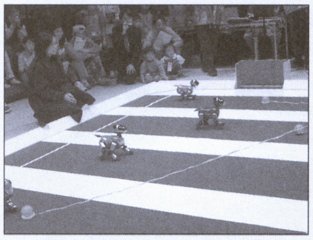


▲筆者がプログラムしたAIBO。ボールを（足元にあるのに）見失って，結果は0点！



▲PK戦で優勝した福岡工大のAIBO。低姿勢でボールを逃さない

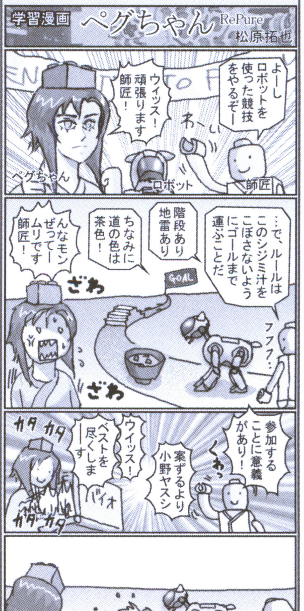
筆者（松原）も自作のプログラムで挑んだのですが，画像処理がうまく機能せず，惨憺たる結果に終わりました。ロボットプログラミングを極めるために，もっと精進したいです。



▲レース競技の様子

超豪華 プレゼント！

参加賞でもらった「AIBOプログラミングメモリースティック（16MB）」と，ジャンケンに勝ってもらった「ネットワークウォークマン＆MagicGate対応メモリースティック（64MB）」(PK戦，3位は同点だったため，賞品はじゃんけんで決定)をプレゼント（詳しくは147ページをご覧ください）



自作一代記

電子工作にチャレンジ!

連載第21回

大人気の「大人の科学」シリーズの最新作『大江戸からくり人形』

製作・文／松原拓也

学研「大人の科学」シリーズ 第8弾『大江戸からくり人形』
価格：5，900円(税別)
学研「大人の科学」WEBサイト
http://kids.gakken.co.jp/kit/otona/index.html

今回は先日発売された，大人気の「大人の科学」シリーズの最新作『大江戸からくり人形』を紹介します。

茶運び人形を忠実に再現

「大江戸からくり人形」は，「からくり半蔵」と呼ばれた細川半蔵頼直という人による「機巧図彙」という技術書に書かれた「茶運び人形」を元にして製作したキットだそうです(**写真1**)。

「茶運び人形」は，江戸時代，主に大名がお客を楽しませるため珍重していた遊び道具とのことですが，当時は最先端の技術を取り入れた贅沢品。今，こうしてキットとして手軽に楽しめるのは，非常に嬉しいことです。

組み立て開始

では早速，組み立ててみましょう(組立て時間は1時間半～2時間とのこと)。

《写真1》「大江戸からくり人形」のパッケージ

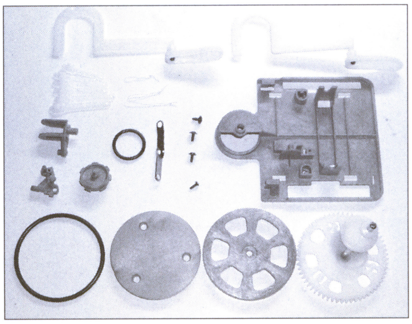
《写真2》付属するパーツ

このキットは，マニュアルが親切に解説しています。さらに，パーツが工程別に袋詰めされていて便利です(**写真2**)。接着剤は不要で，ドライバーとピンセットが付属します。用意する道具はハサミだけです。

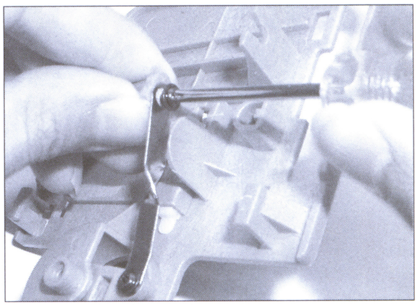
本体の材料はプラスチック製ですが，茶碗は焼き物，着物は布製で，かなり本格的です。

台車の組み立て

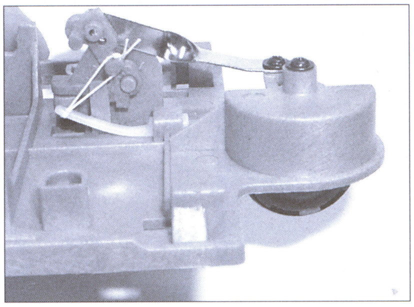
まず最初に，動輪と左右の足を底板に取り付けます(**写真3，4**)。この人形は三輪で移動します。そして，前輪は角度



《写真3》台車を作るためのパーツ



《写真4》付属ドライバーでねじを固定



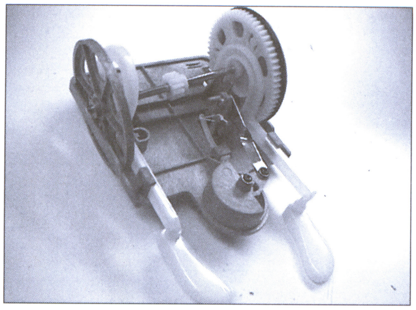
《写真5》人形の進行方向を制御する

を変えることで，人形の進行方向を制御できます。前輪は板バネの力で，まっすぐに保たれていて(**写真5**)，爪が回転板に当たることで可動するように作られています。

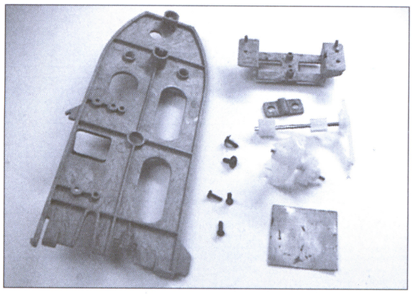
左右の足は，動輪がカムを使って連動していて，「すり足」で歩いているような動作をすることができます(**写真6**)。

テンプの組み立て

「テンプ(天符)」を取り付けます(**写真7**)。テンプとは，動力の速度を調整す



《写真6》台車の完成

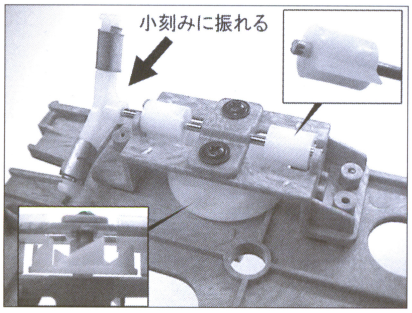


《写真7》テンプ組み立て用の部品

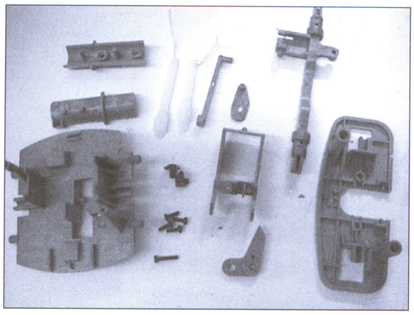
るための部品です。テンプに巻きつけた金属板の重りによって，回転速度を調整できます(写真8)。金属板を切るためには，ハサミが必要です。

テンプ歯車(行司輪と呼ばれる)にはギザギザの歯があり，その部分に二つのストッパーが順番に引っかかることで，回転速度が制御されます(写真9)。この機構は，和時計の調速機を応用したものだそうです。

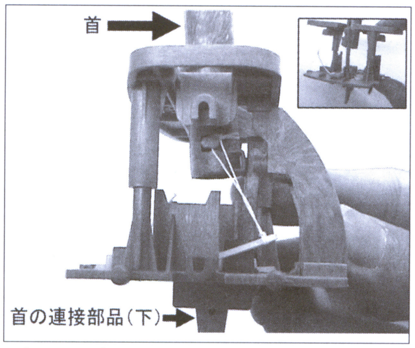
《写真8》テンプに重りを付ける(重さで速度を調整)



《写真9》テンプ歯車を軸受けに固定



《写真10》胸部の組み立て用のパーツ



《写真11》胸部の完成

胸部の組み立て

人形がお辞儀をする仕組みを組み立てます(写真10)。人形の首は板バネで引っ張られることで，垂直に保たれます(写真11)。そして，連結された「爪」に力が加わることで首を振ります。

動力部の組み立て

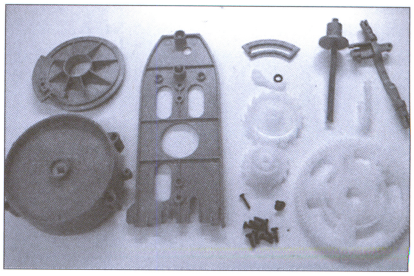
動力部を組み立てます(写真12)。動力となるゼンマイ(写真13)には金属製のスプリングが使われていますが，「機巧図彙」によると当時は原料に「クジラのヒゲ」が使われていたそうです。右側板に，大歯車と回転板，ゼンマイ軸を固定してから(写真14)，先ほど組み立てた底板と左側板を取り付けます(写真15)。

あと，腕の動きが，動力のスイッチとして伝達するように「自動制御棒」という長いクギを取り付けます(写真15)。

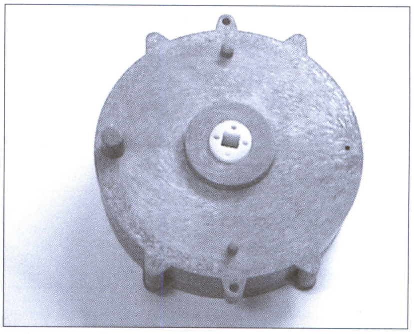
腕・頭の取り付け

腕と頭，そしてお盆を取り付けます(写真16，17)。頭はピンポン玉と同じスチロール製で，軽いです。

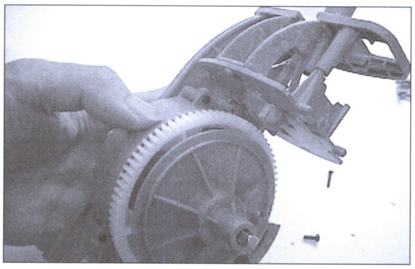
まだむき出し状態ですが，ひとまずこれで完成です(写真18)。



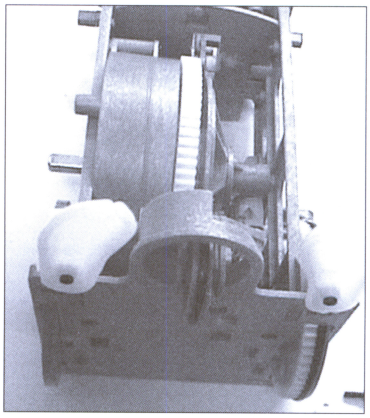
《写真12》動力部のパーツ



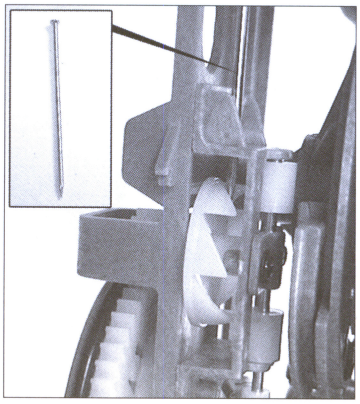
《写真13》動力源となる「ゼンマイ」



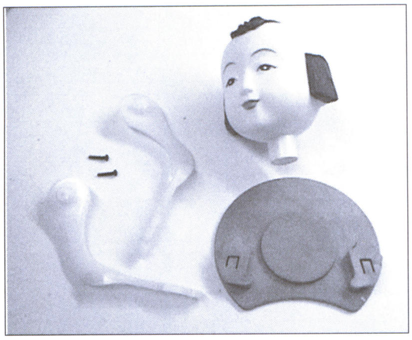
《写真14》右側板に胸部とゼンマイ，ゼンマイ軸を取り付ける



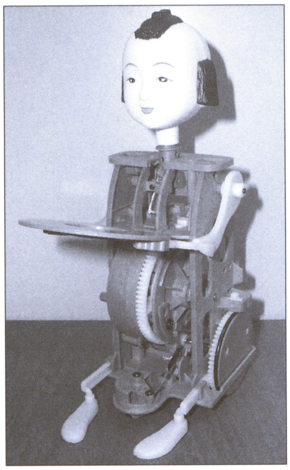
《写真15》底板の取り付け



《写真16》自動制御棒の取り付け



《写真17》腕，頭，お盆のパーツ



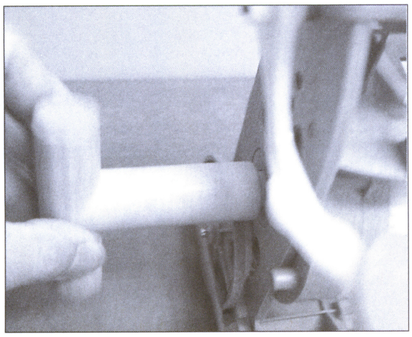
《写真18》ひとまず完成した状態。全高は26cmほど

動作確認

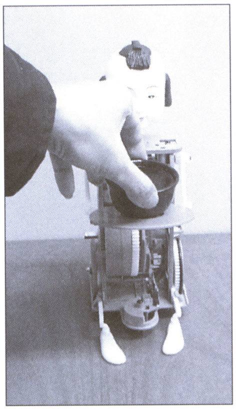
服を着せる前に，からくり人形がちゃんと動くか，確認してみます。

まず，ゼンマイを巻きます(写真19)。次に，人形を床に置き，茶碗をお盆に乗せると，人形が直進し始めます(写真20)。人形が約90cmほど進むと，首を動かしてお辞儀をします。この時，茶碗を取ると，人形の動きが止まります(写真21)。

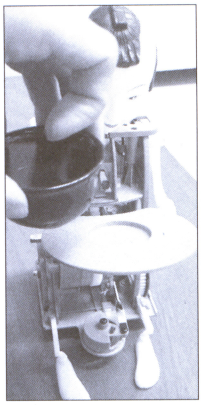
それから，お茶を飲み，空の茶碗をお盆に乗せると，人形が旋回します(写真



《写真19》ゼンマイの巻き上げ。クラッチが動いて「バキバキ」と音がする



《写真20》茶碗を乗せると，「カタカタ」という音をたてて人形が動き出す



《写真21》茶碗を取ると，人形が停止する。お辞儀をしていることにも注目

《写真22》茶碗をお盆に戻すと，Uターンして帰っていく。動きが速い！

22)。向きを変え終ると直進して，元の場所に帰っていきます。お茶碗を取ると人形は止まります。ここで取らないとまたUターンしてしまいます。

旋回する角度などは，各部の調整ねじを回して調節できます(写真23,24,25)。

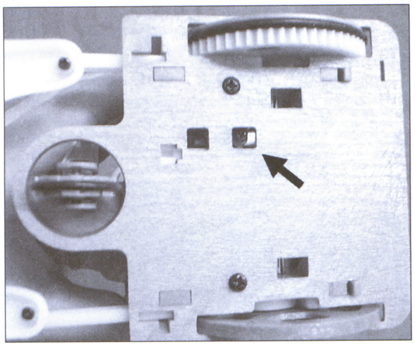
最後に服を着せれば，本当に完成です(写真26,27)。

茶運びのカラクリ

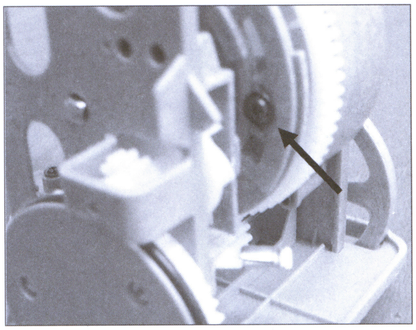
お辞儀をしたり，向きを変えたり，止まったりと，多彩な動きをするこの人形ですが，ここで，この動作原理を考えてみましょう。

まず，茶碗を乗せて動き出すのは，自動制御棒の働きです。自動制御棒はテンプ歯車のストッパーとして機能しています。この棒は，腕が下がることで持ち上がり，ストッパーが外れるようにできています。

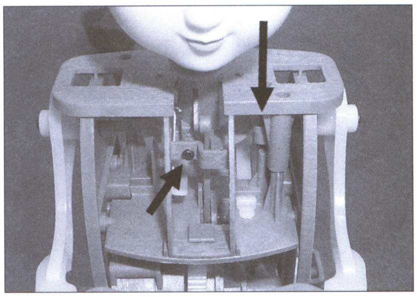
お辞儀と旋回の仕掛けは，ゼンマイ軸の回転板にあります。回転板には，二つの出っ張り(カム)があり，それが，お辞儀と旋回の機構を押し出すように作られています。回転板にはワンウェイクラッチという一方向にしか回転しない機構が組み込まれていて，ゼンマイを巻いた時



《写真23》底板のねじで，走行時の直進性を調整。右に回すと右寄りに，左に回すと左寄りに進む



《写真24》回転板を回す(右回り)ほど直進距離は短くなる。さらに調節駒を右に動かすと旋回の角度が大きく，左に動かすと旋回の角度が小さくなる



《写真25》胸部のねじで首の角度を調整。肩付近のねじで，お盆の水平角度を調整できる

に，回転板が初期位置に戻るようになっています。

あと，足が前後に動いて「すり足」で歩いているかのように見える機構は，車輪隣にあるカムによるものです。カムは回転軸の中心からずれているので，クランク運動をするように作られています。

まさに江戸時代のロボット

「機巧図彙」が書かれたのは，なんと1796年のことです。ゼンマイ駆動や，カム，ワンウェイクラッチなどの技術が，今から300年も前に確立されていたというのは，本当に驚きです。モータこそ無

《写真26》付属の着物。ちゃんちゃんこ，はかま，上着

かったものの，これらは現在のロボット開発に匹敵する技術だったと思われます。

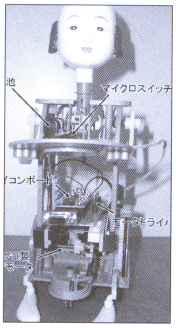
「大人の科学」シリーズは，ちょっと大きな本屋さんなどで簡単に入手できます。これを機に，江戸時代の最新技術に触れてみる，というのはいかがでしょうか。

《写真27》着物を着せて完成！

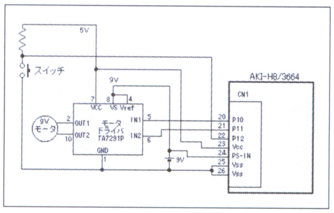
(番外編)茶運びロボットの製作

まるっきり冗談みたいなものですが，「大江戸からくり人形」と，秋月電子通商の「AKI-H8/3664」を使って，「茶運びロボット」を作ってみました(写真28，第1図)。行っている制御は，腕についているマイクロスイッチが押されたらモータを回転させるだけです(リスト1)。

このロボットは電気仕掛けで動く茶運び人形を目指したつもりなのですが，進行方向を変えたり，お辞儀することができません。からくり人形の優秀さを改めて思い知りました。



《写真28》16bitマイコンを搭載した茶運びロボット



《第1図》回路図

《リスト1》茶運びロボット用プログラム(AKI-H8/3664用)

```
/* P12のスイッチが押されたら   */
/* P10-11に接続したモータを回転*/

#include <3664f.h>              /*I/0アクセス用インクルード*/
                                                        /*ファイル            */

#define DIR_FOW  1      /*モータI0:正転*/
#define DIR_BACK 2      /*モータI0:逆転*/

#define POW_MAX         8       /*モータパワー:最大*/
#define POW_MID         4       /*モータパワー:中*/
#define POW_MIN         1       /*モータパワー:最小*/

#define TIME_ON                 30      /*モータを回転させるループ回数*/

/* モータ回転 */
/*      char dir   : 回転方向*/
/*      char power: パワー1-8 */
/*      int time   : 回転時間1- */
void subMot(char dir ,char power ,int time )
{
        #define WAIT_C                  0x200   /*モータオン／オフの最小ループ回数*/
        int cnt;
        int wait;

        if(power > POW_MAX) power=POW_MAX;
        if(power < POW_MIN) power=POW_MIN;

        for(cnt=0 ;cnt<time ;cnt++) {
                IO.PDR1.BYTE = dir;                     /*モータ駆動*/
                for(wait=0 ;wait<(WAIT_C * power);wait++){
                }
                IO.PDR1.BYTE = 0x0;                     /*ストップ*/
                for(wait=0 ;wait<(WAIT_C * (POW_MAX-power));wait++){
                }
        }
        IO.PDR1.BYTE = 0x0;                     /*ストップ*/
}

void main()
{
        IO.PMR1.BYTE = 0x00;                    /*ポート１を汎用ＩＯに設定*/
        IO.PCR1 = 0x03;                                         /*ポート10-11を出力に設定*/
        while(1) {
                while(IO.PDR1.BYTE & 0x4){
                                /*スイッチがOFF(H)の場合はループ*/
                }
                subMot(DIR_FOW ,POW_MID , 30);     /*正転/
        }
}
```


AKI-PIC プログラマーキット Ver.3
シミュレータソフト付のすぐれもの!
最新PICフラッシュマイコン版、ほぼ全てに対応!
PIC16F84 マイコンモジュールが直接焼けます!
●PIC16C5X ●PIC16F83/84 ●PIC17CX ●PIC12C508(509)
●PIC16F874・877 ●PIC16C6X対応 ●Windows98対応CD-ROMソフト付
●マイクロチップ社CD-ROMソフト付 1キット¥6,700
★マイクロチップ社のPIC(ピック)マイコンのほとんど全てがプログラマできます★28ピンゼロプレッシャソケット・40ピンゼロプレッシャソケット付★RS232C(ADM232AAN)パソコン通信回路内蔵★当社オリジナル日本語対応アセンブラ、CD-ROM付★ウィンドウズ98版ライターコントロールソフト付(シミュレータソフト付)★専用ガラスエポキシ基板使用★マイクロチップ社製品カタログCD-ROM付★テスト用PIC12C509、PIC16F84-10/P(10MHzレゾネータ付)付
専用電源セット¥1,000 15V 0.8A ACアダプタ+9P Dサブコード1m

AKI-H8/3048F(フラッシュメモリ内蔵)超高速動作 16MHz版(20MHz可)超高性能マイコンボード
★H8/3048F(日立)を使った超高速・超高集積ボードです。
★100ピンH8CPUとRS232C ICは基板実装済みで高信頼。
★外形寸法はAKI-80と同一サイズ5×7cm。
★ガラスエポキシ・スルーホール加工ボードです!
多量販売により、お求めやすい価格が実現! 1キット¥3,500
AKI-H8開発キット(即使えるキット)
20P(10×2)ソケット×1、10P(5×2)ソケット×1、16×2 LCDモジュール付 ¥7,800
(開発ソフト、アッセンブラーソフト、ライター一式付)、技術(ハード)資料集付、ソフト資料集付、ボード操作、アプリケーション集付、専用マザーボード付、BASICコンパイラソフトCDROM付/DC19V 30Wスイッチングボード(電源)付/H8/3048F用CコンパイラCD-ROM付

AKI-H8/3052F(512KBフラッシュメモリ) 超高速25MHz版マイコンボード
★H8/3048Fの4倍(4Mバイト)の大容量。H8/3048Fとほぼ同一ピン(5V書込み)同じソフトで動作します。★H8/3048Fボードと同一外形、同一ピン配置、主要部品全て実装済み★ソケット&ピンソケットどちらにも対応できます。ピンヘッダ40ピン×2連×2個、20ピン×2連(40P)ソケット×2個、10ピン×2連(20P)ソケット×1個、5ピン×2連(10P)×1個 付★H8/超高速書込み(ターボライター)ソフト付き。
H8/3048 16MHz、H8/3067 20MHz、H8/3052 25MHz、H8/3069 25MHzに対応しています。 1セット¥4,200 CDROMソフト、取扱い説明書付

超小型軽量カラーCCDカメラ基板ユニット MTV-54CON
★ミラーファンクション(左右反転機能)内蔵★1/4インチCCD 25万画素(512H×492V)★レンズ:3.7mm F2.0 画角68°★解像度:380本★最低照度:1Lux
★電源:12V 65mA
★基板サイズ 26×22mm 高さ30mm
重さ11.6g
1台¥8,750 専用ケーブル DCジャック、コネクタ配線済

高感度CCDカメラ基板ユニット
(台湾製 完成品 完全調整済み)赤外・モノクロ両用タイプ★フリッカレス回路内蔵★有効画素数512H×492V★可変電子シャッター+オートアイリス★1Vp-pビデオ出力★電源DC12V 120mA
MK-0343E 1台 ¥3,500
シャープ製CCD使用
赤外投光器付(LED6個実装済)

PIC BASICモジュール 20MHz動作 A/Dコンバータ内蔵
★PIC16F877に整数型BASICインタプリタ本体が書き込まれています。★付属の専用ソフトでエディット、デバック、書き込みまでできます。232C通信なども専用のコマンドで簡単に使えます。
PIC BASICモジュール基板、専用ソフト(WIN98、ME,XP対応)CD付 1セット¥2,700

AKI-PICベーシック開発セット 新発売
★AKI-PIC877ベーシックモジュール+専用ベース基板(9P RS232Cコネクタ、1024K E²PROM付)★16×2LCDユニット★8LEDモニター付 1セット¥5,400 LCD、LED専用コントロールボード一式付

PICマイコン制御 ステッピングモータドライブキット
★正転・逆転OK ★8ビットマイコン制御で使い易く、精密安定にコントロール可 ★技術資料、専用ボード、C.R、パワーTR一式付
※モータ無しキット 1キット¥1,200

PICマイコン制御 ステッピングモータドライブキット
★7.5度(48ステップ)¥400 ステッピング付キット★データ付 1キット¥1,500

PICマイコン制御 ハイパワーステッピングキット
★1.8度(200ステップ)¥700モータ付 ★データ付 1キット¥1,700

2相48ステップ小型ステッピングモータ
★48ステップ/相(7.5度)
★軸3mm★外形4.2×2.3cm★重量110g
¥400技術資料付

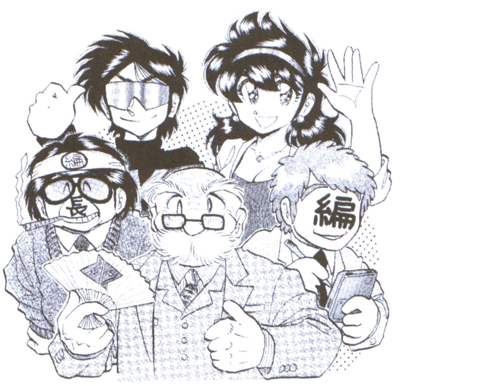
2相200ステップ精密ハイパワーステッピングモータ
★200ステップ/相(1.8度)
★軸8mm★外形5.5×5cm★重量350g
¥700 技術資料付

AKI-PIC F877-20／ICスタンプ
★20MHzレゾレータ、セラコン、LED付40ピン(20+20)ソケット ★PIC16F877のフラットICをDIPに変換したIC 1コ¥900
PICベーシック書込済品 ¥1,400

PICフラッシュマイコンIC
PIC16F628-20/P 1コ¥300、10コ¥2,800
PIC16F84A-20/P 1コ¥380、10コ¥3,500
PIC16F873-20/SP(28ピンスリムDIP) 1コ¥800
PIC16F874-20/P(40ピンDIP) …… 1コ¥800
PIC16F876-20/SP(28ピンスリムDIP) 1コ¥900
PIC16F877-20PT(44ピンフラットIC) 1コ¥800
AT90S2313-10 1コ¥380、10コ¥3,400

(株)秋月電子通商 http://akizukidenshi.com
http://www.akizuki.co.jp からもご覧になれます。
秋葉原本店 〒101-0021 東京都千代田区外神田 1-8-3 野水ビル 1F/2F
TEL.03-3251-1779 FAX.03-3251-3357
代引通販で、さらにラクラクお買い物!! インターネット専用・通販センター
〒334-0063 埼玉県川口市東本郷252 TEL.048-287-6611 FAX.048-287-6612

あくせすROOM



編集部ニュース

影： 来月から，あくせすROOMでの採用者（毎月1名）に，編集部からのささやかなプレゼントをしよう。3月号で掲載された人には，なんと「平成14年お年玉クジつき年賀状」の束を提供するのだ。ほしい人はハガキの表（住所を書く側）に，「平成去年の年賀葉書くれ」と書くべし。

編： ちなみに抽選ではないです。

つぐ美： 年末大掃除のあとの大処分大会ってことね。

新コーナーのお知らせ

影： 過去に文字ネタを中心とする投稿者として誌上で活躍した新歓1005円氏を知っているかな？　つい先日まで，彼がインターネット上でいろいろお題を出して，参加者がボケた回答をするという企画があったのだ。その名も「微喜利（ビキリ）」。寄席でおなじみの「大喜利」を一字変えただけだが，ボケまくりの参加者と，素でボケまくりの司会者による絶妙なコンチェルトは，ちょっとよそでは味わえないだろう。ところが，ISP（インターネットの接続業者）に継続手続きを忘れてたりして，微喜利は突然消滅。もったーねーという多くの要望をかんがみて，ベーマガ誌上にてこの企画を継続することにしたのである。まあ，ぶっちゃけて言えば，文字ネタ要素の強い旧・読者の闘技場みたいなものか。

編： 第1回は4月号（3月8日）です。そこで，2月15日必着で，みなさんの絶大なる投稿葉書を募集します。あんまり文字量が多すぎると載せにくいけど，葉書をそのまま載せてしまうこともあるかも。読みやすくて恥ずかしくない文字で書いてくださいね。

①新学期の「自己紹介」。 クラスの注目間違いなし！ってヤツを教えて頂戴。例・プロモビデオ配布

②人生相談コーナー
ていうか，みんなの日頃抱えている悩み相談をぶつけるコーナーだ！
そして解決方法をオイラがまったりと受け流す！（流してどーする！）

影： 気軽に投稿してもらいたいということで，ベーマガ公式サイトからの電子メール投稿も受付中。
下記にアクセス。ハガキ代が浮くぞ！
http://www.basicmagazine.net/bikiri/

つぐ美： しかも，毎月1名様に賞品がつくそうです。これは試作品が完成しだい発表の予定です。

「ベーマガとの出会い」

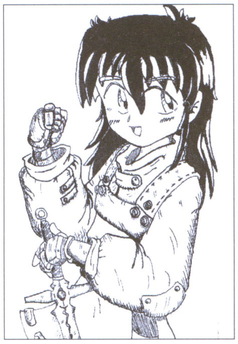
※ありえなさそうな話でも実話です。
　02年春，某所の古本屋でPC関連の本をあさっていたところ，貴社出版の「電脳遊技考」（90年発行）を見つけたのですが，何気なく眺めていると，見慣れた名前が…(P.371)参照
「桜井 政博」さん…ご存じの方もいるかも知れませんが，彼は「大乱闘スマッシュブラザーズ」シリーズ（共に任天堂）のディレクターなのです。（90年当時，なんとHAL研入社前!!）彼はゲームに対する意見を多面的かつ熱く投稿していました。桜井さんのゲームに対する打ち込みようは前から彼のHPで知っていた（実は書き込みもしました！）のですが，10年も前からこんなに熱くゲームについて語る人がいたなんて！とても感動しました。そして自分もプログラム・ゲームに興味があったので，ベーマガを読ませて頂くことになったのです。
　あの時から10年余…どうやらゲームについての議論熱は冷めたようですねー。しかし，ベーマガが桜井さんのような名クリエータを輩出したことを知ったので，自分も頑張れば，いっぱしのクリエータになれるかも知れないと思いました。そのためにもライターさんの方々，皆さんも頑張って下さい！
　多少（いや，かなり）長くなってスミマセン。

◀石川県珠洲市・銀狐

Dr.D： 歴史が長いだけに，ベーマガ読者から数多くのクリエイターが生まれておるし，クリエイターのタマゴがベーマガを盛り上げてきたものじゃ。注意しなくてはイカンのが，ベーマガを教科書的に読んだだけで，業界で活躍できるわけではないということじゃな。

編： ベーマガは授業というよりは，部活動だと思っています。プログラムやツールを使いこなす楽しみを知り，予期しないバグなどを克服する気持ちを育て，一緒に磨きあう友人を見つける場なんですよ。

つぐ美： コンピュータ業界に直接関係がない会社でも，ちょっとVisualBsaicやVBAの知識があるだけで，仕事がガンガンはかどるってことが多いのよね。

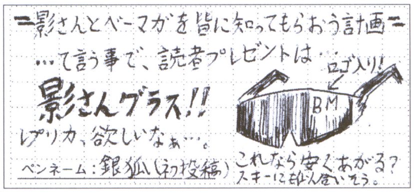


▲神奈川県平塚市・らうま

編： ひと仕事終えたあとのガッツポーズでしょうか。

影： これは練習用の剣でしょ。

つぐ美： 動きにくいカッコウで練習してると，正しい形がくずれちゃうわよ。



▲石川県珠洲市・銀狐

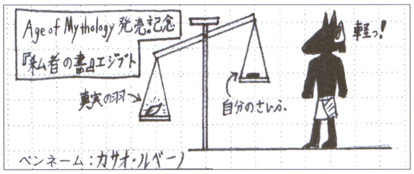
影： これはすばらしい。ついでにロゴ入りのマスクもつけようか。

編： 99円ショップで売っているサングラスにシールを貼って完成。

▲福島県伊達郡・レギオンさん

Dr.D： しかるべき研究所に就職するといいぞ！

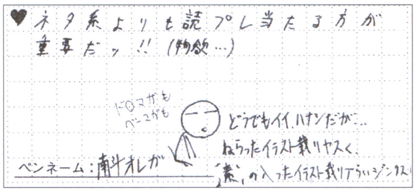
編： ウイルスを防ぐには，ウイルスを作れる知識が必要なのも現実です。



▲茨城県ひたちなか市・カサオ・ルベーノ

影： 現金は持ち歩かない主義らしい。

編： 『Age of Empire』のファンタジー世界版。遊んでますか？



▲福島市・南斗オレガ

影： プレゼントによって応募者がかたよるので，そこがねらい目。

編： 今月は「スクールウォーズ」のDVDビデオに人気集中。

あくせすROOMの参加方法

あくせすROOMでは，みなさんの投稿をお待ちしています。愛読者カードや，官製ハガキ，または同サイズの丈夫な紙などを封書でまとめるなどして，お送りください。
鉛筆や色鉛筆はきれいに印刷されません。せっかくペンで描いてあっても，鉛筆の下書きが残っているものは，やはり載りにくくなりますので，インクや水性ボールペンなど，はっきりと書ける道具を使いましょう。文字だけの投稿であれば，ワープロを使ってもかまいませんよ（そのまま掲載できます）。

応募先
〒141-8715　東京都品川区東五反田1-11-15　電波新聞社マイコンBASICマガジン編集部「あくせすROOM」係

2月号お楽しみプレゼント

①PIC-BASIC書き込み済み「AKI-PIC877ベーシックモジュール」（CD-ROM付属）／10名（提供：秋月電子通商）
②PIC-BASIC書き込み済み「AKI-PIC 16F877-20/ICスタンプ」／20名（提供：秋月電子通商）
③「クロコダイル実験室：物理・理科編」アカデミック版／2名（提供：アスキーソリューションズ）
④「AIBOプログラミングメモリースティック（16MB）」／1名（提供：編集部）
⑤「ネットワークウォークマン＆MagicGate対応メモリースティック（64MB）」／1名（提供：編集部）
●応募方法：本誌とじ込み愛読者ハガキ（またはインターネット読者ハガキ）にプレゼント記入欄があります。一つだけ選んでお送りください。
●締め切り：2003年2月7日必着
《2002年12月号の当選者》①神奈川県相模原市・中村恒平さん　②大阪市・多田雅一さん他9名　③群馬県前橋市・小幡倫行さん他2名

Falcom® NEWS EXPRESS

日本ファルコム株式会社
〒190-0012 東京都立川市曙町1-14-13
TEL 042-527-6501
FAX 042-528-2714

スタッフ募集中
採用担当BFN係
http://www.falcom.co.jp/

緊急速報！

西風のラプソディに続く，ファルコム海外プロデュース第2弾！

月影のデスティニー
月影のDESTINY

2003年3月 発売決定！

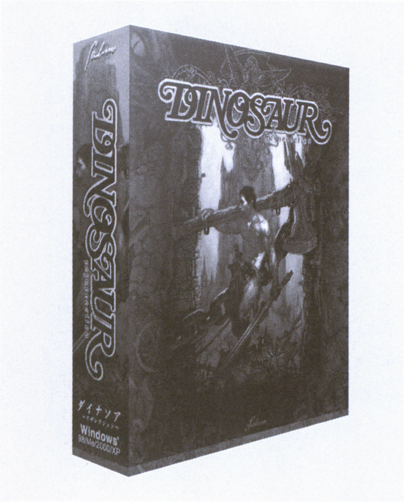
年明け早々ビッグニュースだぞ！ファルコム海外プロデュース第2弾の発売が決定したのだ。第1弾は，そう「西風のラプソディ」。これまで，日本ではアジアのゲームは売れないという常識を覆し，発売日に即日完売する店が続出。その後家庭用ゲーム機にも移植されるなど，異例の大ヒットになったわけだけど，そのファルコムが自信を持ってお送りするのが，この「月影のデスティニー」。ファルコムだから実現したChina英雄伝説，武侠アクションRPGの決定版が，ついに日本上陸だ！

それでは「月影のデスティニー」がどんなゲームなのかちょっとだけ紹介するぞ！

●驚かされるのはキャラの表現力と，美しい情景！
髪をかきあげる仕草や，肩を震わせて泣くシーンなど，キャラクター達の表現力は実に緻密で多彩！鮮やかな色彩と渾身のディテールで描かれた美しく雄大な景観はまさに絶景！

● アクション性の高い戦闘と新感覚武功システム！
月影のデスティニーの目玉とも言える戦闘は，走って，跳んで，カンフー映画のように画面狭しと動きまわり，敵の攻撃をかわしつつ剣で斬りつける爽快感！武功システムによる育成の楽しみも加わって，常に新鮮な戦闘を味わえる！

「ダイナソア～リザレクション～」第3の結末とは！？

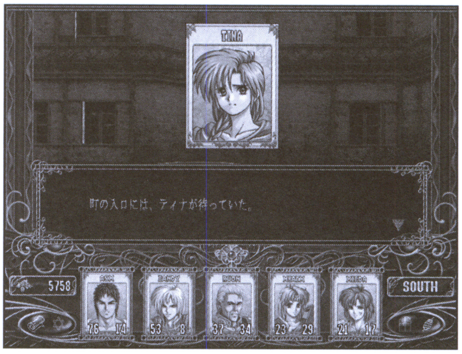


ファルコムホームページでも公開中なので，ぜひご覧ください。

http://www.falcom.co.jp

ダイナソア～リザレクション～
Windows98/Me/2000/XP
6大特典付き初回限定版　好評発売中！

ダイナソア～リザレクション～には裏ストーリーがあるのは，もう知ってるよね。パーティキャラも，表のキャラより個性が強いので，ぜひプレイしてみてね。ところが，その裏ストーリーに，実はもう1つの結末があるってことは知ってますか？初版のオリジナル版にはなかった結末なのでぜひ見て欲しい！といっても，どうすればいいかわからないと思うので，ヒントを1つ。まず，裏ストーリーに行くこと。一度クリアして最初から開始すると，蝶のイベントがいきなり始まり，その蝶を追いかけなければ，裏へ行くことができるぞ。そして，ストーリーを進めていくうちに，いくつか選択肢が出るシーンがあるけど，普通はこうするよねという選択の逆を選ぶのだ。大サービスで，1つめの選択は，右の写真がヒント。あと，もう1つあって，その選択を選んだことによってゲームオーバーにならなければ，第3の結末にいけるぞ。

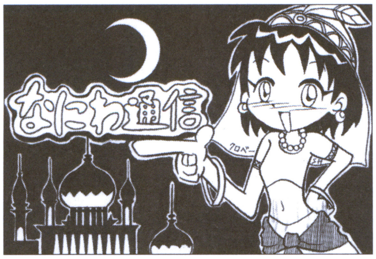


ダイナソア～リザレクション～着メロデータ配信中！

ファミ通.comの着メロ配信サービスで「ダイナソア～リザレクション～」の下記3曲が現在配信中！
「失われしものたち」「精霊の賛歌」「FRONT LINE」 ●アクセス方法はiモード端末の「iメニュー」から「エンターテイメント」→「ゲーム」→「ファミ通.com」です。

ファルコム本続々刊行！

ダイナソア～リザレクション～の攻略ガイドが，角川書店とソフトバンクパブリッシングの2社から発売されるのは，先月号でもお知らせしましたが，発売日も1/18に決定！完全攻略ガイドなので，裏シナリオもばっちりと紹介されているぞ！そして，1月下旬には，宙出版から，「アンソロジーコミック イースオールスター4コマ+（プラス）」という，イース全シリーズのパロディ4コマが発売されるのだ。また，同じく宙出版から「VM JAPANビジュアルブック（仮）」，新紀元社からは，英雄伝説ガガーブトリロジーの，シナリオに沿った解説本「英雄伝説ファンブック（仮）」も発売予定だ。白き魔女，朱紅い雫，海の檻歌すべてを掲載。何と500ページ以上の超ボリュームとなるみたいだぞ。詳細が決まり次第，ファルコムホームページでも，どんどん紹介していくのでお楽しみに！



癒し系？ なにわ通信

タイトル冠文：大阪府・中ノ島金太郎
タイトル・イラスト：長崎県・クロベー
本文：暁

PRESENTED BY マイコンソフト
http://www.micomsoft.co.jp

ADDRESS● 〒530-0005大阪市北区中之島3-2-4
朝日新聞ビル6F電波新聞社大阪本社内
マイコンソフト株式会社 なにわ通信係

新年，明けましておめでとうございます。

お正月には，お年玉をたくさん貰えましたでしょうか?

さて，マイコンソフトでは，新年から新製品の開発で大忙しです。

前号でもお知らせしましたD端子セレクターの開発も佳境となり，社内は慌ただしくなってまいりました。

今回も，新製品のD端子セレクターに関する，情報をお伝えしましょう。

D端子で行こう

開発中のD端子セレクターは，基本的にはD端子映像と音声の切り替え機なのですが，下記の機能も備えています。前号の記事でもふれましたが，今回はもっとディープに説明していきたいと思います。

D端子映像をRGBへ

本機にはD-Sub15ピンのRGB出力端子が付いているため，入力されたD2とD4の映像を，パソコン用ディスプレイに表示することができます。

プログレッシブ出力が可能なDVDプレイヤーのD2映像(525p)や，任天堂さんのGAMECUBEのD2映像をパソコン用のディスプレイでも表示できます。

また，ハイビジョン放送でデータ放送等に使用されているD4(750p)の映像もパソコン用ディスプレイで表示することが可能です。(ただし，本機には映像のガンマ補正機能などを備えていないため，家庭用テレビなどよりも画面は暗く表示され，薄暗い部分は，はっきりと表示できない可能性があります。)

また，ハイビジョンで一般的に使用されておりますD3(1125i)の映像は，RGB方式に変換することはできますが，インターレース(飛び越し走査)と呼ばれる表示方式なので，残念ながら，パソコン用のディスプレイでは表示できません(パソコン用ディスプレイはノンインターレス方式のみ動作を保障しているためです)。

尚，本機には，アップスキャン機能等は内蔵されておりませんので，D1(525i)の映像はパソコン用ディスプレイでは表示できません。

コンポーネントを正しいD端子へ

本機には，コンポーネント端子(Y,PB,PR)の信号入力が2系統(D端子と共有)用意されており，コンポーネント端子をD端子に変換することができます。

D端子は，基本的にコンポーネント端子の端子形状を変更しただけのものです。

このため，市販の「コンポーネント端子をD端子に変換するケーブル」を使用すれば，コンポーネント端子の映像をD端子付きテレビに表示できるはずです。

しかしながら，テレビの中にはD端子特有の信号を読み取って画面を表示している機種があり，変換ケーブルだけでは表示できないことがあります。(例えば，シャープさんの一部のテレビ等)。

本機では，こういったテレビでも問題なくコンポーネント端子の映像を表示できるように設計されておりますのでご安心ください。

もし，お手持ちのコンポーネント端子付きDVDプレイヤー等の映像がD端子付きテレビに表示できなくて困っておられる方は，本機を使用すると正しく表示できるようになるかもしれませんね。

マルチ21ピン端子付き

本機には，マルチ21ピンのRGB入力端子を備えており，旧型のRGB出力を持つゲーム機をD端子でテレビに接続できます。

また，マルチ21ピンのRGB端子は，変換ケーブルを利用することで，Y,PB,PR端子としても使用できますので，RGB入力を利用しない方でも端子が無駄になりません。

付属ケーブルは？

本製品には，本機とテレビを接続するための「D端子ケーブル」，「ステレオ音声ケーブル(RCA仕様)」，「コンポーネント-RGB変換ケーブル」を，各1本ずつ付属しております。

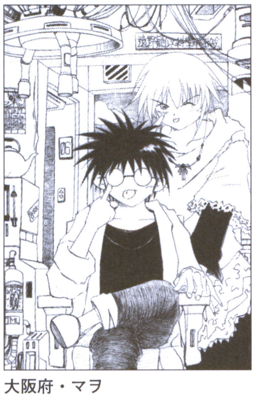
また，リモコンも付属しております。

発売予定

さてさて，前号に引き続き，セレクターの機能を詳しく説明させていただきましたが，新製品の魅力をご理解いただけましたでしょうか?商品名や製品写真は次号で発表できると思います。皆さんの，お年玉を有効に活用するためにも，是非，本セレクターの購入をご検討ください。

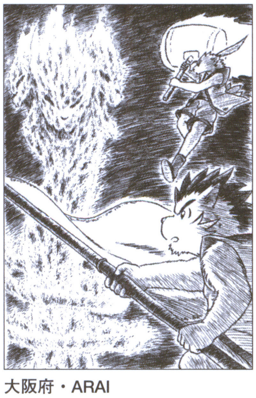
発売は，2003年2月を予定しております。

ありがたい おハガキのコーナー



大阪府・マヲ
アッカンベーをされてしまいました(笑)。この研究所では，アッカンベーの心理的効果について研究しているのかもしれませんね。

愛知県・邪道坊や
ワイルドな感じのキャラクタですね。高倉健は私も好きです。
寡黙なところがなんとも良い感じですよね。



大阪府・ARAI
「火消しの武流守」シリーズがスタートしました。「悪魔の火」を退治する，火消しの姉弟のお話だそうです。次回も期待してます!

AQUAPLUS公認

「P/ECE HAND BOOK」

好評発売中！

まかべひろし 著　　定価：本体1,800円＋税

■A4変型/本体128頁/B5判開発リファレンス48頁/12cm CD-ROM1枚付属

本書の内容

●これがP/ECEだ!●開発者突撃インタビュー!●アプリケーションカタログ●初心者のためのアプリケーション開発解説●開発Tips●ハードウェアスーパー改造テクニック●必ず役立つ!P/ECE関連Webサイト選

「P/ECE HANDBOOK」専用サイト　http://www.jc2.co.jp/p/piecetop.html

ISBN 4-88554-716-4

●お求めは全国の書店，または弊社出版販売部へ

電波新聞社　〒141-8715　東京都品川区東五反田1-11-15　電話（03）3445-8201（ダイヤルイン）

ISBN 4-88554-727-X

AQUAPLUS公認

「P/ECE HAND BOOK」 VOL.2

好評発売中！

まかべひろし 著　　定価：本体1,500円＋税

■A4変型/本体128頁/本誌掲載データ完全収録 12cm CD-ROM1枚付属

本書の内容

●アクアプラス突撃インタビュー●未発表Leaf新作!●ハイクオリティソフト全ソース収録!●雑誌連合本音座談会!●佐藤裕美さん、P/ECEで声優初挑戦!●松田浩二のドット絵講座●クリエーターへの道をつかもう!

「P/ECE HANDBOOK」専用サイト　http://www.jc2.co.jp/p/piecetop.html

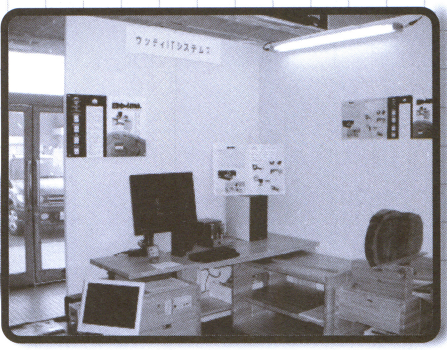
●お求めは全国の書店，または弊社出版販売部へ

電波新聞社　〒141-8715　東京都品川区東五反田1-11-15　電話（03）3445-8201（ダイヤルイン）

日本人ならやっぱり"木"です！

「CODAMA」

木製パソコン机ブランド 木霊



▲「技術力ぐんま 2002 ー中小企業技術振興展ー」にも出展。好評で多くのお客を集めた

仕事，勉強，プライベート……。パソコンはずっと身近なものになり使用時間は飛躍的に増加しています。しかし長時間，パソコンを扱っているとイライラしてきませんか。じつはそれはパソコンの画面だけでなく，周囲の環境のせいかもしれません。イライラを解決し，人にやさしく，しかも周囲にやさしいパソコン環境を実現するために生まれたのがウッディITシステムズの「CODAMA」(木霊）です。

「CODAMA」は天然国産材を使用し，伝統家具のデザインと機能を持つ人に優しい木製パソコン机ブランドです。日本人の生活に深く関わってきた"木"を取り入れることでパソコン環境に安らぎと落ち着きを与えてくれます。

また従来のパソコン机はスチール製のものがほとんど。木製の机にパソコンを置こうと思えば置けますが，配線の問題など不便な問題が発生します。しかし「CODAMA」は「パソコンを使う」という発想からスタートした木製机。パソコン設置スペースや配線も考えた作りとなり，見た目にも暖かく洗練されたパソコン環境を演出します。

デスクタイプ，座卓タイプ，サイドデスクタイプ，キャビネットタイプ，DOS/Vパソコン内蔵型キャビネットタイプ，またこれらの複合型（**写真1**）など多彩なラインナップと組み合わせが可能。家庭，オフィスにぴったり合った商品を選ぶことができます。

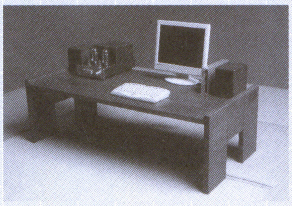
とくに座卓タイプ（**写真2**）は，今まで和室にいまいちフィットしなかったパソコンをばっちり和風にアレンジ。畳の部屋にも違和感なく溶け込みます。背面コーナ部に設けられたケーブルホール(**写真3**）によって配線もすっきり。パソコン机としての利便性も備えます。

またDOS/Vパソコン内蔵キャビネット（**写真4**）は，内部のパソコンはいつでもバージョンアップが可能（**写真5**)。機器の進歩に合わせてパソコンの性能をアップさせ，キャビネットはずっと使い続けることもでき，家具として長い付き合いができます。

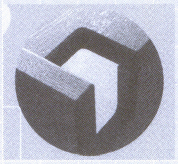
これらの商品はスチール製より約3割高い価格帯となっており7－12万円。木材の状況などによって多少価格が変動しますので，価格などくわしくは下記販売代理店までお問い合わせください。

《写真1》サイドデスクタイプ，PC内蔵型キャビネットなどを複合させた「ITスタンダードフルセット」

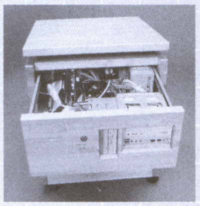
《写真4》DOS/Vパソコン内蔵型キャビネットタイプ「CP-2500/3500」



《写真2》和室にぴったり座卓タイプの「ZD-9000/12000」



《写真3》座卓でのPC使用を考慮したケーブルホール



《写真5》内蔵型キャビネットは，中味のパソコンをいつでもバージョンアップできる親切設計

ウッディITシステムズの成り立ち

「CODAMA」ブランドを作り出すために生まれた異業種集団ウッディITシステムズ。きっかけはパソコン工房前橋店のマイダックス（群馬県前橋市）の井田光則取締役が清水製函（群馬県高崎市，清水一廣取締役）製の家具を見かけ，「木のパソコン環境を作り上げたい」と持ちかけたことから。

その後，清水製函，マイダックスに家具製造販売の須賀木工製作所（群馬県高崎市，須賀徹代表取締役），組立器検査機製造販売のシグマエンジニアリング（群馬県玉村町，斉藤悟代表取締役）を加え，4社がそれぞれの得意分野を持ち寄って木製ITシステム環境を作り上げました。

群馬県4社によって結成されたウッディITシステムズの「CODAMA」ブランドは，群馬県によって平成13年，"ぐんまブランド創出事業"に認定されています。

販売代理店

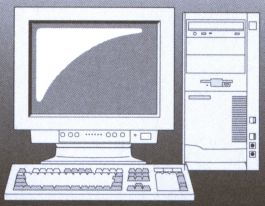
（有）清水製函
〒379－0031高崎市上大類町364
TEL：027－352－3331
FAX：027－352－3758

（株）須賀木工ショールーム
〒370－0018高崎市新保町1067
TEL：027－363－4781
FAX：027－363－7309

パソコン工房前橋店
〒379－2147前橋市亀里町396－2
TEL：027－290－3163
FAX：027－290－164
URL：http://www.midax.co.jp

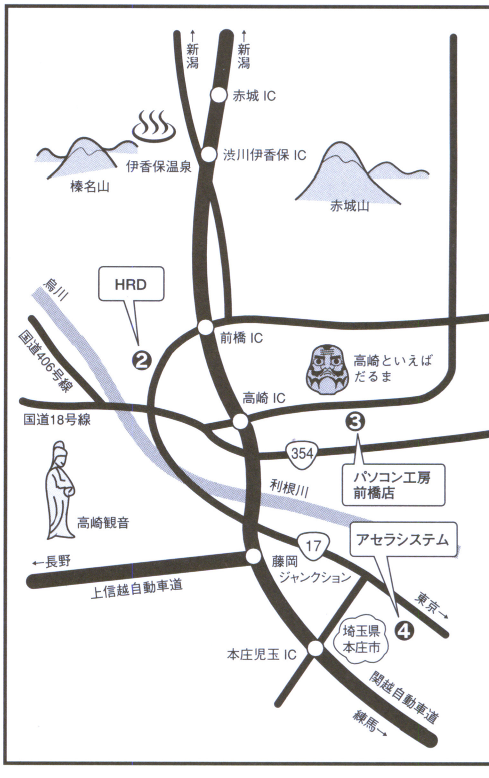
CODAMA

北関東地区 DOS/



「マイコンBASICマガジン」おすすめショップ

北関東のDOS/Vショップは今日も元気に頑張っています！
女性店長あり、話好きの店主ありと個性豊かな店ばかり。
変わらないのはみんなパソコンが好きだということ。
一度、近くのショップに立ち寄ってみては。
親切丁寧に応対してくれますよ。



HIT＆RUN

http://www.leoplanet.co.jp/hit/

全国中古部品高価買取中！

●指定の運送会社が引取りに伺います。（買取価格￥3500円超）

STEP1. 参考買取価格を調べる。
STEP2. 買取申込（メールかFAX）
STEP3. 梱包し，発送準備する。
STEP4. 指定日に運送便訪問
STEP5. 査定価格連絡メール
STEP6. 良ければ代金支払

ショップ営業時間
AM10:00～PM7:30（火曜定休）

●お問い合わせ、ご注文はインターネット、又はTEL、FAXにても受付けております。
●通信販売での支払法
銀行振込、ヤマト便の代金引換払いにてお願いします。
●メールマガジン（週刊）"DOS/V価格ウォッチ"
購読者募集中です。

〒372-0813 群馬県伊勢崎市韮塚町561-1
☎ 0270-21-1491 FAX 0270-26-9561
http://www.leoplanet.co.jp/hit/

取扱い品

●オリジナルPC組立キット（各種）
●組立費￥6,000より，OSインストール費￥2,000より
●ノートパソコン修理，HDD換装
●各種DOS/Vパーツ全般

中古ノートPC販売例

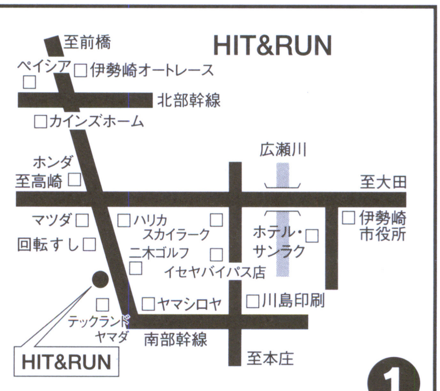
●K6-2-300/64MB/4.3GB/CD24X モデム，USB×1，13"DSTN Dynabook：￥53,800-
●PentiumⅡ233/96MB/3.2GB 12.1TFT LCD SVGA/USB×1 Fujitsu FMV6233：￥39,800-

中古部品 高値買取り

●各社マザーボードとビデオカード他
●各種ドライブ類他 ●ノートPC，周辺機器

ノートPC HDDグレードアップサービス

HDD交換サービス（見積りご相談下さい）
●2.5インチHDD代金
（10GB：￥10,500- 20GB：￥12,500- 30GB：￥15,000-）
＋
●交換取付料 ￥6,000-
＋
●OS＆データ転送料 ￥7,000-

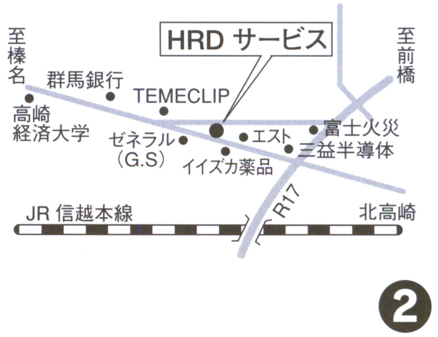


自作パソコン＆パーツのお店
HRDサービス

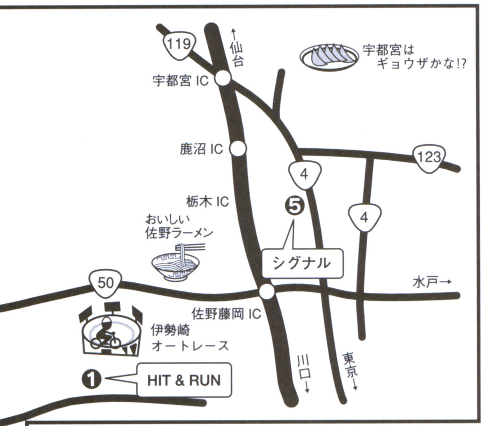
インターネットで最新情報をGET！
http://www.atm-japan.co.jp/hrd/

●メンバーズカード会員募集中!! 会員特価でお得!!
●中古ジャンクコーナー充実！買い取りも行います！

〒370-0801 群馬県高崎市上並榎町479-4 ☎ 027-370-3700



Vショップマップ



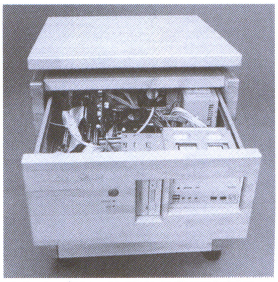
ハンドメイドパソコンショップ

パソコン工房 前橋店

安さ・信頼・技術 DOS/V パーツの一番店

AM11:00～PM8:00　水曜定休　☎027-290-3163
〒379-2147　前橋市亀里町396-2
http://www.midax.co.jp

ATXパソコン内蔵型キャビネット



ぐんまブランド創出事業選定製品

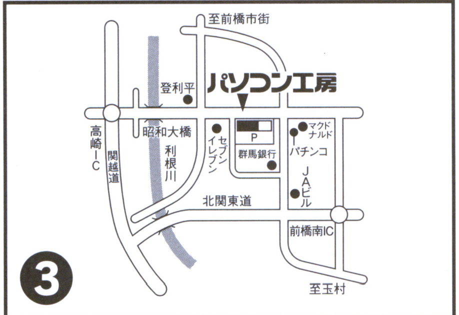
無料パソコン組立教室

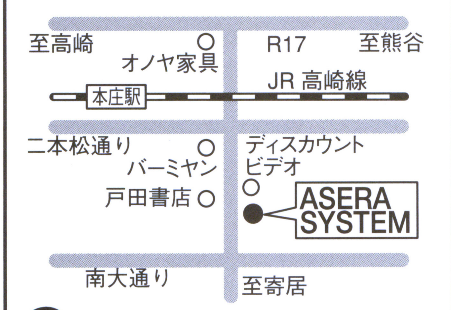
毎週月，金，日（要予約）

中古パソコン 部品買取り，販売します

好評リサイクルパソコン販売中
LinuxPC ¥7,800～　中古ノートPC 等

パソコン修理，改造，グレードUP致します





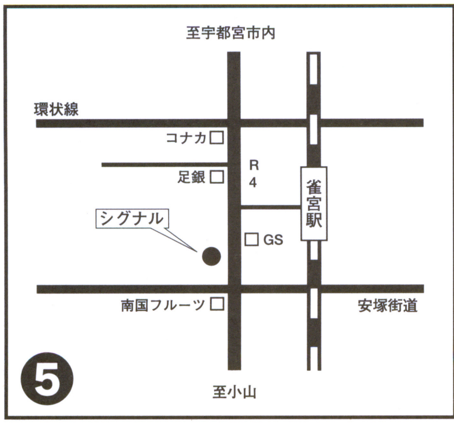
ASERA SYSTEM

ASCAL

挑戦　DIYPC
ご相談下さい！PCトラブル
親切・丁寧・安心サポート
（土）（日）の午後ーご相談受け付けてます。
パーツ・組立PCのことなら当店へ

DOS/Vアセラシステム

埼玉県本庄市北堀866-1　Tel/Fax 0495-23-0379
（営）AM10:30～PM7:30（火曜定休）
HP移行しました。http://asera.chat.-jp.com/



シグナル

http://www.signal-pc.com

「Cube型ベアボーンキット 好評発売中！」

最新情報はホームページで！

営業時間
10:00～19:00（日曜休、祭日営業）

〒321-0132
宇都宮市雀の宮4-5-12
☎028-655-2983
FAX 028-655-2969
e-mail:signal@signal-pc.com

中古通販

PC NET
ピーシーネット

在庫保有数日本最大級
(常時10,000台)

TEL.03-3733-1441

通販営業時間／月～金 9:00～19:00 土・日・祝日 11:00～19:30

●アドレスが新しくなりました

www.pc-net.jp/

今までのアドレスでも引き続きご利用いただけます。
www.prins.co.jp/

楽天市場店 http://www.rakuten.co.jp/pcnet/ ネットプライスモール http://www.netpricemall.com/
ビッグローブショッピングワールド http://shopping.biglobe.ne.jp/

急げ!!

最新情報は電話かネットでチェック!!

ピーシーネットのお年玉サービス

1月2日～31日まで **送料無料!!**

御注文の際は、必ず「雑誌を見た!!」を合い言葉で!

合い言葉が無い場合は、送料は有料となるので注意!

※キャンペーン・セール商品は省く、通常価格商品のみ対象とさせて頂きます。

インターネットでは毎回お得な企画がいっぱい!!

NEW! りょうのコレで決めた!

期間限定タイムセール

お支払い方法 代金引換・銀行振込・各種カード可能

全商品 外装クリーニング 無料! サービス

[安心の全品保証付] 1万円以上…1ヵ月保証、1万円以下…1週間保証

※本商品は添付品（OSを含む）は付属しておりません。
※中古パソコン販売時の在庫定義が変わりました。
今後在庫とは、従来品に加え、動作上支障のないキズ・天板やけ・カバー等の欠品がある物や、キーボード等が補充（他メーカー含）されている商品も、「在庫」としてお取り扱いさせていただきます。

DOS/V

注文No.	商品	価格
U10866	Aptiva E(2190-24J)(IBM)(K6-2(450)/64MB/6GHD/CD40/外装ヤケ/外装塗装剥げ/省スペース型/PCIX3/USBX2)	¥18,800
U10846	FMV-6600CL4E(富士通)(PenIII(600)/128MB/10GHD/CD/外装ヤケ/省スペース型)	¥36,800
U10807	NETVISTA M41(6792-45J,46J)◆未開封新古品(IBM)(PenIV(2.2G)/256MB/40GHD/CD48/未開封/PCIX3/USBX2/AGPX1)	¥119,800
U10805	NETVISTA A21(6339-83J)◆新古品・未開封(IBM)(Celeron(1G)/64MB/20GHD/CD48/未開封/PCIX3/USBX4/付属品付)	¥73,800
U10803	NETVISTA A22(6339-1JJ)◆未開封・新古品(IBM)(Celeron(1.10AG)/64MB/20GHD/CD48/未開封/PCIX3/USBX4)	¥76,800
U10806	NETVISTA A21(6339-PBJ)◆未開封・新古品(IBM)(PenIII(1G)/128MB/20GHD/CD48/未開封)	¥73,800
U10809	NETVISTA A40P(6841-GAJ)◆未開封・新古品(IBM)(PenIII(933)/128MB/30GHD/CD48/サーバー/未開封)	¥63,800
U8594	PC-MA26D/C-42(NEC)(PenII(266)/128MB/4.3GHD/CD24/外装ヤケ/擦り傷/スタンド無し/省スペース型/PCIX4/USBX2)	¥15,800
U10863	PC-MA50J/SZ…85(NEC)(PenIII(500)/128MB/8.4GHD/CD32/外装ヤケ/外装塗装剥げ/PCIX3/USBX2)	¥27,800
U10732	PC-MA56H/C(NEC)(Celeron(566)/64MB/10GHD/CD40/外装ヤケ/省スペース型/PCIX2/USBX2)	¥32,800
U10837	PC-MA10T/E(NEC)(PenIII(1G)/256MB/40GHD/CD-R/RW/省スペース型/PCIX3/USBX4/AGPX1)	¥69,800
U10590	DESKPRO EN C333/3.2(COMPAQ)(Celeron(333)/64MB/3.2GHD/CD/外装ヤケ/擦り傷/外装塗装剥げ/省スペース型/PCIX1/共有X1/USBX2)	¥13,800
U10832	DESKPRO EN 6500/10(COMPAQ)(PenIII(500)/128MB/10GHD/CD32/外装ヤケ/擦り傷/外装塗装剥げ/PCIX2/ISAX1/共有X2/USBX2/AGPX1)	¥26,800
U10841	OPTIPLEX GX110(DELL)(PenIII(733)/64MB/20GHD/CD/外装ヤケ/省スペース型/PCIX2/USBX2)	¥39,800
U10759	Dimension C(DELL)(PenIII(933)/256MB/20GHD/CD/外装ヤケ/擦り傷/省スペース型/PCIX2/USBX2)	¥52,800
U10503	GATE WAY GP7-700◆Win98SE付き(GATEWAY)(PenIII(700)/128MB/20GHD/CD/外装ヤケ/ミドルタワー型/高さ:48cm)	¥44,800
U10821	ESSENTIAL◆WinME付き(GATEWAY)(PenIII(800)/128MB/46GHD/CD-RW/省スペース型)	¥49,800
U10848	SELECT L7-1000(GATEWAY)(Athlon(1G)/64MB/45GHD/CD/スロットカバー無し(2)/ファイルカバー無し(2)/ミドルタワー型/高さ:48.5cm)	¥64,800
U10851	SELECTJP/OX KER◆WinME付き(GATEWAY)(Athlon(1.1G)/256MB/30GHD/CD/RW/外装ヤケ/スロットカバー無し/WinME/ミドルタワー型/高さ:45cm)	¥69,800
U10751	PERFORMANCE(GATEWAY)(PenIII(933)/256MB/30GHD/CD/MO(動作未チェックの為保証無し)/外装ヤケ/ミドルタワー型/USBX2)	¥59,800
U10678	Vectra VLI8/600SF(HP)(PenIII(600)/64MB/8.4GHD/CD/外装ヤケ/高さ:9cm)	¥29,800
U8573	FMV-C40L6◆液晶セット(富士通)(Celeron(400)/64MB/8.4GHD/CD24/15TFT(1024X768)/省スペース(液晶セット)/PCIX2/USBX2)	¥41,800
U10011	FMV-3CLH101.161.151(富士通)(PenIII(450)/64MB/6.4GHD/CD24/省スペース型/縦・横共通/PCIX2/USBX2)	¥21,800
U10093	FMV-4CLK121,111,181,171(富士通)(Celeron(500)/64MB/10.2GHD/CD24/外装ヤケ/省スペース型/PCIX2/USBX2)	¥25,800
U10855	FMV-3DXJ101,161,151(富士通)(PenIII(500)/64MB/6.4GHD/CD32/外装ヤケ/擦り傷/MO付き(動作未チェックの為保証無し)/モデム無し/PCIX2/ISAX2/USBX2)	¥24,800
U10812	FMV-SL1K101(FMV600SL)(富士通)(Celeron(900)/128MB/40GHD/CD/外装ヤケ/外装塗装剥げ/キーボード無し/マウス無し/省スペース型/高さ:34cm)	¥62,300
U10860	FLORA 330 PC1DC6-(A,E,B)6012HC00(日立)(Celeron(366)/32MB/4.3GHD/CD24/14.1TFT(1024X768)/外装ヤケ/液晶セット(省スペース)/PCIX2)	¥36,800
SH960	CF-6862T3J◆Win98付き◆今だけのお買い得商品!!(パナソニック)(PenII(400)/64MB/6.4GHD/CD32/付属品付/PCIX2/ISAX1/共有X1/USBX2)	¥21,800

DOS/Vノート

注文No.	商品	価格
U20400	ThinkPad 235(2607-15J)(IBM)(MP(233)/96MB/3.2GHD/SVGA/外付FD付き/外装ヤケ/外装塗装剥げ/9.2TFT)	¥54,800
U20286	ThinkPad 240X(2609-51,52J)(IBM)(Celeron(450)/64MB/12GHD/外付けFDD付き/リアカバー無し/FDコネクタカバー無し/10.4TFT/B5サイズ/USBX1)	¥69,800
U6718	ThinkPad 310E(2600-FOJ)(IBM)(MP(166)/48MB/2.1GHD/SVGA/擦り傷/ゴム足無し/12.1TFT)	¥14,800
U6504	ThinkPad 380ED(2635-6A,YJ)共通(IBM)(MP(166)/32MB/3GHD/CD20/SVGA/擦り傷/外装ヤケ/外装ヒビ/12TFT)	¥24,800
U20889	ThinkPad 390E(2626-B0J,B5J)(IBM)(Celeron(366)/32MB/4.8GHD/CD24/SVGA/CD・FD着脱式/12.1TFT/A4サイズ/USBX1)	¥65,800
U20918	ThinkPad 390X(2626-F(0.5.N)J)(IBM)(Celeron(400)/64MB/6.4GHD/CD24/SVGA/外装ヤケ/擦り傷/リアカバー無し/12.1TFT/A4サイズ)	¥64,800
U20922	ThinkPad X20(2662-34J,35J)(IBM)(PenIII(600)/128MB/20GHD/XGA/LAN/FDドライブ外付け(USB接続)/外装ヤケ/キーボード文字剥げ/12.1TFT/A4サイズ/USBX2)	¥108,900
U6685	ThinkPad 570(2644-1(A,D,B)7J)(IBM)(PenII(333)/64MB/4GHD/CD24/XGA/ウルトラベース付き/擦り傷/13.3TFT/A4サイズ/USBX1)	¥59,800
U20557	ThinkPad 600E(2645-4A,T,L,K,J)(IBM)(PenII(366)/32MB/6.4GHD/CD24/XGA/外装ヤケ/擦り傷/外装塗装剥げ/AC型番違い/13.3TFT/USBX1)	¥66,800
U6828	ThinkPad Iシリーズ(2611-455)(IBM)(MP(300)/64MB/4.8GHD/CD24/XGA/AC型番違い/ゴム足無し/液晶キズ有り/13.3TFT)	¥59,800
U20942	ThinkPad A20M(2628-F1J,F2J)(IBM)(PenIII(750)/128MB/20GHD/CD24/XGA/LAN/15TFT/A4サイズ/USBX1)	¥109,000
U20919	ThinkPad A21M(2628-G(1.2)J)◆Win98付き(IBM)(PenIII(800)/192MB/20GHD/DVD/XGA/LAN/外装ヤケ/擦り傷/拡張カバー無し/12.1TFT/A4サイズ/USBX1)	¥128,800
U20921	ThinkPad X23(2662-ESJ,ERJ)(IBM)(PenIII(866)/128MB/30GHD/CD24/XGA/FDドライブ外付け(USB接続)/外装ヤケ/12.1TFT/A4サイズ/USBX1/拡張ステーション付き)	¥138,000
U20275	PC-VA23C/WXA…23(NEC)(MP(233)/32MB/2.1GHD/CD20/XGA/LAN/外装ヤケ/13.3TFT/USBX1)	¥34,800
A0134	PC-VA40H/WSTAN47(NEC)(Celeron(400)/64MB/4.3GHD/CD24/SVGA/トップカバー傷/12.1TFT/USBX2)	¥64,800
U20700	PC-VA43H/WS…68(NEC)(Celeron(433)/64MB/6GHD/CD24/SVGA/LAN/外装塗装剥げ/擦り傷/12.1TFT/A4サイズ/USBX2)	¥68,800
U20916	PC-LC500J34DR(NEC)(PenIII(500)/64MB/12GHD/XGA/CD-RW/モデム/外装キズ/外装塗装剥げ/14.1TFT/A4サイズ/USBX2)	¥79,800
U20804	PC-VA80J/WX(NEC)(PenIII(800)/128MB/20GHD/CD-RW(未チェックの為保証無し)/モデム/XGA/14.1TFT/A4サイズ/USBX2)	¥118,000
U20784	ARMADA 7400(COMPAQ)(PenII(300)/128MB/6GHD/CD24/XGA/英語版キーボード/擦り傷/ゴム足無し/PCカバー無し/13.3TFT/USBX1)	¥49,800
U20940	LATITUDE CPI D233ST(DELL)(PenII(233)/64MB/3.2GHD/CD24/SVGA/PCダミーカード無し/キーボード文字剥げ/12.1TFT/A4サイズ)	¥34,800
U20735	INSPIRON 7000(DELL)(PenII(366)/128MB/6.4GHD/CD/擦り傷/PCカバー無/15/A4サイズ)	¥59,800
U6050	FMV-NP131,132,133(FMV-BIBLO NP13)(富士通)(MP(133)/32MB/1.6GHD/CD8/SVGA/モデム/外装ヤケ/擦り傷/ゴム足無し/12.1TFT/拡張ステーション付き)	¥24,800
U6630	FMV-NE36X1.3.6(富士通)(Celeron(366)/64MB/10GHD/CD24/XGA/擦り傷/13.3TFT/USBX1)	¥56,800
U20929	FMV-6NUGXC2.0.6.5(富士通)(Celeron(450)/64MB/6GHD/CD24/XGA/LAN/外装キズ/13.3TFT/A4サイズ/USBX2)	¥67,800
U20325	FMV-NE660C2.3◆WinME付き(富士通)(Celeron(600)/64MB/20GHD/CD24/XGA/14.1TFT)	¥96,800
SH957	FMV9NURLC6◆Win98付き(富士通)(Celeron(750)/128MB/20GHD/CD24/14.1TFT/A4サイズ/付属品付)	¥106,800
U20872	Tornado 3136(Akia)(PenII(366)/64MB/6.4GHD/CD24/SVGA/PCダミーカード無し/擦り傷/12.1TFT/A4サイズ/USBX1)	¥56,800
U6346	PAP301JC(東芝)(MP(266)/96MB/4.3GHD/FAXモデム/SVGA/擦り傷/外装塗装剥げ/PCダミーカバー無し/外付けFDD付き/10.4TFT/USBX1)	¥32,800
U6517	DynaBook PA1270C9(東芝)(PenII(266)/64MB/4GHD/CD20/XGA/外装ヤケ/13.3TFT/USBX1)	¥46,800
U6729	DynaBook PAS206N8.N5.NW.NN.NK(東芝)(K6-2(366)/64MB/4.3GHD/CD24/SVGA/LAN/外装キズ/12.1TFT/USBX1)	¥53,800
U6711	PAS255N5,NS,NK,N8,NW,NM(東芝)(Celeron(366)/64MB/6.4GHD/CD24/LAN/14.1TFT/USBX1)	¥59,800
SH959	DynaBook Satellite2210(PA-SA50C2C8)(東芝)(Celeron(500)/64MB/6GHD/CD24/SVGA/LAN/12.1TFT/A4サイズ/USBX1)	¥84,800
V0244	PA-SA50C4C8(Satellite2210)◆Win98付き(東芝)(Celeron(500)/64MB/6GHD/CD24/XGA/100TX/14.1TFT/付属品付)	¥86,900
U20946	FLORA 210(PC-5NLO02-ID5DA,DE)(日立)(MP(233)/32MB/3.2GHD/SVGA/外付けFDD付き/擦り傷/9.2TFT/A4サイズ/USBX1)	¥29,800
U20856	FLORA 270SX(PC1NH8-H4D22K310)(日立)(Celeron(333)/64MB/4.8GHD/CD24/LAN/XGA/外装擦り傷/外装塗装剥げ/外装ヒビ有り/13.3TFT/A4サイズ/USBX2)	¥47,800
V0240	PC7NW1-GDE23H120◆優良商品!!◆Win98SE付き(日立)(Celeron(550)/64MB/6GHD/CD24/100TX/XGA/14.1TFT/A4サイズ/付属品付)	¥87,800
SH961	CF-L2P8C2S◆Win2000付き(パナソニック)(PenIII(850)/128MB/30GHD/CD24/XGA/LAN/13.3TFT/A4サイズ/USBX2/付属品付)	¥148,000
U20005	MEBIUS PC-PJ10MZ(シャープ)(Celeron(333)/64MB/8.1GHD/CD24/XGA/PCダミーカード無し/外装ヤケ/14.1TFT/USBX2)	¥59,800
U20944	VAIO PCG-767◆Win98付き(SONY)(MP(266)/192MB/4GHD/CD24/XGA/モデム/擦り傷/リアカバー無し/12.1TFT/A4サイズ)	¥54,800
U20927	PCG-XR7Z/K(PCG-8624)◆Win2000付き(SONY)(PenIII(750)/192MB/20GHD/DVD/XGA/外装塗装剥げ/PCダミーカード無し/14.1TFT/A4サイズ/USBX1)	¥136,600

バッテリーパックは消耗品につき保証対象外とさせていただきます。

98デスクトップ＆ノート

注文No.	商品	価格
U50056	PC-9821V233/M7C2.D2(NEC)(MP(233)/64MB/3GHD/CD20/外装擦り傷/マウス無し/PCカードリーダー付き/ミドルタワー型/PCIX2/USBX2)	¥14,800
U8610	PC-9821RA223/N30(NEC)(PenII(233)/64MB/3GHD/CD16/LAN/外装ヤケ/外装キズ/PCIX2)	¥14,800
U2644	PC-9821RV226/N20(NEC)(PenII(266)/32MB/2GHD/CD6/外装ヤケ/擦り傷/外装塗装剥げ/ミニタワー型)	¥54,800
U8996	PC-9821RA300/M40.W40.D40(NEC)(Celelon(300)/64MB/4GHD/CD24/LAN/外装ヤケ/擦り傷)	¥69,800
U3572	PC-9821LS150/S1C2(NEC)(MP(150)/32MB/1GHD/CD10/SVGA/外装ヤケ/擦り傷/AC型番違い/キーボード文字剥げ/バッテリーパック無/12.1TFT)	¥27,800
U2766	PC-9821NA12/H8(NEC)(Pen(120)/16MB/810HD/CD/PCM/バッテリーパック欠品/外装ヤケ/10.4TFT)	¥9,800
U3557	PC-9821NA15/X14(NEC)(Pen(150)/16MB/1.4GHD/CD8/PCM/外装ヤケ/外装ヒビ/バッテリーパック無し/12.1TFT)	¥32,800
U6143	PC-9821NR13/S14R(NEC)(Pen(133)/32MB/1.4GHD/SVGA/外装ヤケ/擦り傷/CD付き/12.1TFT)	¥17,800
U3630	PC-9821NR15/S10(NEC)(Pen(150)/16MB/1GHD/CD無し/キーボード文字外装ヤケ/CDドライブ欠品(外付けケーブル含む)/外装ヤケ/擦り傷/12.1TFT)	¥12,800
U50047	PC-9821NR150/S20(NEC)(MP(150)/48MB/2.1GHD/SVGA/外装塗装剥げ(パームレスト)/擦り傷/12.1TFT)	¥17,800
U6052	PC-9821NR166/X30F(NEC)(MP(166)/32MB/3GHD/CD10/XGA/モデム/擦り傷/外装ヤケ/13.3TFT)	¥39,800
U6806	PC-9821NR233/S32A(NEC)(MP(233)/32MB/3.2GHD/CD16/SVGA/擦り傷/12.1TFT)	¥59,800
U6463	PC-9821NW133/D14C.D(NEC)(MP(133)/32MB/1.4GHD/CD11/モデム/SVGA/擦り傷/キーボード文字剥げ/12.1DSTN)	¥17,800

バッテリーパックは消耗品につき保証対象外とさせていただきます。

Macintosh

注文No.	商品	価格
U9480	PowerMac G3/DT233(Apple)(PPCG3(233)/96MB/4GHD/CD24/外装ヤケ/擦り傷/KB別売)	¥25,800
U9361	PowerMac G3/DT266(Apple)(PPCG3(266)/32MB/4GHD/CD24/外装ヤケ/KB別売)	¥26,800
U9584	PowerMac G3/MT266(Apple)(PPCG3(266)/256MB/4GHDX2/CD/外装ヤケ/擦り傷/KB別売/ミドルタワー型)	¥35,800
U9574	PowerMac G3/400(ブルーマック)(Apple)(PPCG3(400)/192MB/6GHD/DVD(動作未チェックの為保証無し)/擦り傷/外装ヤケ/KB付き)	¥59,800
U9563	PowerMac G3/450(ブルー)(Apple)(PPCG3(450)/256MB/9GHD/CD/外装ヤケ/擦り傷/マウス・電源ケーブル型番違い/KB無し/ミドルタワー型)	¥69,800
U9585	PowerMac G4/500(グラファイトカラー)(Apple)(PPCG4(500)/512MB/27GHD/DVD-RAM/ZIP付き(未チェックの為保証無し)/外装ヤケ/擦り傷/KB付き/USBX2)	¥118,000
U40017	PowerBook G3/233(Apple)(PPCG3(233)/96MB/2GHD/CD/XGA/FDD(オプション品)付き/14.1TFT/A4サイズ)	¥63,800
U40018	PowerBook G3/233(Apple)(PPCG3(233)/64MB/2GHD/CD/FDスーパーディスク付き/外装キズ/*FDDはオプションタイプ/12.1DSTN/A4サイズ)	¥49,800
U6896	PowerBook G3 SERIES(Apple)(PPCG3(266)/64MB/4GHD/CD/擦り傷/外装塗装剥げ/14.1TFT)	¥66,800
U40005	PowerBook G3/333(Apple)(PPCG3(333)/192MB/4GHD/CD/擦り傷/ゴム足無し/14.1TFT)	¥74,800
U40022	PowerBook G3/400(Apple)(PPCG3(400)/128MB/6GHD/DVD(動作未チェックの為保証無し)/擦り傷/AC型番違い/FDDはオプションタイプの為付属致しません/14.1TFT/A4サイズ)	¥86,800

キーボードは別売となります。

ディスプレイ

注文No.	商品	価格
U60034	F14T63-DF(NEC)(14.1TFT:1024X768max/擦り傷/外装ヤケ/3列15pin)	¥25,800
U4759	JC-1578VM(NEC)(15マルチ:1280X1024max/外装ヤケ/3列15pin)	¥5,900
U4865	M4551(Apple)(15.1TFT:1024X768max/液晶若干外装キズ有り/3列15pin)	¥35,900
U60061	Apple STUDIO ディスプレイ(Apple)(17マルチ:1280X1024max/3列15pin)	¥24,800
U4920	1400FP(DELL)(14TFT:1024X768max/擦り傷/3列15pin)	¥24,800
SH953	6558-43N(IBM)(21マルチ:1600X1280max/3列15pin)	¥24,800
U60030	9514-B06(IBM)(14.5TFT:1024X768max/外装ヤケ/3列15pin)	¥25,800
U4745	L360(NANAO)(15TFT:1024X768max/24～61kHz/外装ヤケ/3列15pin)	¥32,800
U60070	CPD-E220(SONY)(17マルチ:1600X1200max/外装ヤケ/擦り傷/3列15pin)	¥12,800
U60059	VL-1540S(富士通)(15TFT:1024X768max/外装ヤケ/3列15pin)	¥31,800
U4820	FMV-DP97X7(富士通)(17マルチ:1600X1200max/外装ヤケ/擦り傷/3列15pin)	¥10,800
U4955	RDT-141(三菱)(14TFT:1024X768max/外装ヤケ/3列15pin)	¥25,800
U4639	RD-17MX(三菱)(17マルチ:1280X1024max/外装ヤケ/擦り傷/3列15pin)	¥11,800
U4431	A701G(飯山電機)(17マルチ:1600X1200max/外装ヤケ/3列15pin/5BNC)	¥12,800
U4937	A201H(飯山電機)(22マルチ:2048X1536max/外装ヤケ/3列15pin)	¥29,800
U4902	S901G(飯山電機)(19マルチ:1600X1200max/外装ヤケ/擦り傷/3列15pin)	¥19,800
U60040	TSSA 46633JT(飯山電機)(18.1TFT:1280X1024max/外装ヤケ/3列15pin)	¥67,800
U60043	RT150EX(AKIA)(15TFT:1024X768max/外装ヤケ/3列15pin)	¥29,800
U60042	CF-L45(パナソニック)(15TFT:1024X768max/外装ヤケ/3列15pin)	¥31,800
U4339	VX900(GATEWAY)(19マルチ:1600X1200max/外装ヤケ/3列15pin)	¥17,800

プリンタ

注文No.	商品	価格
U5180	PR-D700XX(NEC)(ドットインパクトプリンター:A3/外装ヤケ/擦り傷)	¥44,000
U5299	PR-J760(NEC)(カラーインクジェット:ハガキ～A4サイズ)	¥8,900
U5292	PM-770C(エプソン)(カラーインクジェット:A4対応～ハガキ/外装ヤケ/外装キズ)	¥8,600
U5233	PM-800C(エプソン)(カラーインクジェット:A4対応～ハガキ/外装キズ)	¥9,600

その他

注文No.	商品	価格
U8149	GT-7000U(エプソン)(イメージスキャナ(カラー))	¥9,800
V0068	H-1021(アイドルジャパ)(マウス:DOS/V用 PS2)	¥580

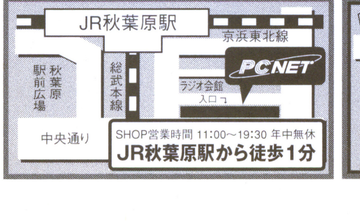
トナー等は消耗品につき標準添付されておりません。別途、ご確認下さい。

商品の状態により価格の変動がある事がございます。詳しくはTELかネット上でご確認ください。

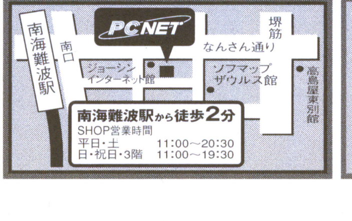
万一、品切れの際はご容赦ください。（12月6日現在）

こちらの在庫リストは通販専用となります。店舗在庫は各店舗へご来店の上ご確認ください。

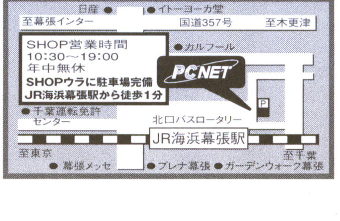
秋葉原駅前店
千代田区外神田1-15-6 ミツウロコビル4F
TEL.03-5209-6111



大阪日本橋店
浪速区難波中2-1-7 なんば東ビル2階・3階
TEL.06-4396-1401



幕張店
千葉県千葉市美浜区ひび野1-9



パシフィックネット

株式会社パシフィックネット
本社 〒144-0051 東京都大田区西蒲田7-10-1
相互西蒲田ビル TEL.03-3733-1441 FAX.03-3733-1440
通販営業時間／月～金 9:00～19:00 土・日・祝日 11:00～19:30

計測機器を各ショップのみで販売中。
詳しくは各ショップまでお問い合わせください。

iモードでショップ最新情報検索
http://www.prins.co.jp/i/

◆ 定期購読●直接購読のしおり ◆

弊社刊行の雑誌は，ご自宅直送方式の年間定期購読をお勧め申し上げます。スピーディーで確実，そしてお得な定期購読をぜひご利用下さいませ。
また，ご希望の雑誌・書籍が，ご近所の書店にない場合は右の振替用紙にて直接ご注文下さい。

■ 年間定期購読料（消費税込料金となっております。）

電　気　店	￥16,000	マイコンBASICマガジン	￥6,120
電波新聞縮刷版（CD-ROM版）		￥96,000	

■ 年間購読の特典

①特別号も通常の定価でご購読いただけます。
②電気店は13冊分の購読ができます。

◆確実な振替口座を利用してご送金下さい◆

送金手数料が不要で，確実な郵便振替でご送金をお願い申しあげます。手続きは簡単です。右の郵便振替用紙に下記の必要事項を記入し，郵便局窓口に必要金額と共にご提出いただくだけですみます。

①購読申込書に必要事項を記入ください。なお1年間，或は2年間購読かをはっきりご記入ください。
②払込通知票，払込票に送金合計金額と住所，氏名をご記入ください。払込通知票と払込票は絶対に切りはなさないようにしてください。なお，必ずフリガナをご記入下さい。
③弊社への御意見・御希望がございましたら，通信欄へご記入下さい。
④すべて記入が終了しましたら，案内文のみ切り離して郵便局窓口に金額を添えて提出してください。

※振替によるご送金は，着本までに約2週間ほどかかりますのでお含みおきください。
お急ぎの場合は，現金書留に，右購読申込書を同封してご送金ください。

この受領証は、郵便局で機械処理をした場合は郵便振替の払込みの証拠となるものですから大切に保存してください。

ご注意
この払込書は、機械で処理しますので、口座番号及び金額を記入する際は、枠内に丁寧に記入してください。
また、下部の欄（表面及び裏面）を汚したり、本票を折り曲げたりしないでください。
（郵政事業庁）

払込取扱票

02 東京

通常払込料金 加入者負担

口座番号（右詰めにご記入ください）

00150-3-51961

金額 千 百 十 万 千 百 十 円 ※

加入者名　株式会社　電波新聞社

料金　特殊取扱

通信欄

※ 定期購読誌名	購読期間	送金額（送料当社負担）
*新規 継続	年　月号より　年間	円
御希望誌・ソフト・グッズ名	冊数	送金額（送料共）
		円
		円

フリガナ
送付先住所
□□□-□□□□

産業分類　所属又は職種

フリガナ
電話 TEL　年令 才　性別 男・女

払込人住所氏名（郵便番号）※
（電話番号）

受付局日附印

各票の※印欄は、払込人において記載してください。

裏面の注意事項をお読みください。郵政事業庁（私製承認 東 第1005号）

切り取らないで郵便局にお出しください。

記載事項を訂正した場合は、その箇所に訂正印を押してください。

払込票兼受領証

口座番号　00150-3-51961（右詰めにご記入ください）　通常払込料金加入者負担

加入者名　株式会社　電波新聞社

金額 千 百 十 万 千 百 十 円 ※

払込人住所氏名 ※

料金（消費税込み）円

受付局日附印

特殊取扱

切り取って郵便局へお出し下さい。

電波新聞社出版物一覧

（消費税込料金となっております）

2002.10現在

最新エレクトロニクス用語事典	5,300円	〒380円	パソコン認定試験合格対策試験問題集3・4級	2,100円	〒310
半導体・ICのすべて	2,520円	〒310円	シスアド受験スーパー記憶術	2,900円	〒340
センサーのすべて	2,000円	〒310円	基本情報受験スーパー記憶術	2,835円	〒340
新デジタル映像技術のすべて	2,100円	〒310円	初級シスアド受験マニュアル	3,780円	〒450
ビデオ技術のすべて	2,100円	〒310円	基本情報受験マニュアル	3,780円	〒450
テレビジョン技術のすべて	2,100円	〒310円	デジタル1種〈技術科目〉最新テキスト	2,205円	〒340
増補改訂版デジタル放送がわかる本	1,600円	〒310円	デジタル3種受験マニュアル	2,940円	〒340
増補改訂版・新ビデオ技術ハンドブック	2,500円	〒310円	アナログ・デジタル全資格〈法規科目〉	1,995円	〒340
新・まんが電気学入門	1,500円	〒310円	特殊無線技士受験マニュアル・2002年版	2,940円	〒340
新電気学入門講座	2,050円	〒310円	はじめよう! インターネットと電子メール	1,575円	〒380
よくわかるDVD&ホームシアター	1,680円	〒310円	Javaプログラミング徹底入門〈基礎編〉	2,940円	〒380
よくわかるプラズマテレビ(PDP)	1,680円	〒310円	情報処理の基本〈デジタル数学編〉	2,625円	〒340
よくわかるデジタルテレビ	1,680円	〒310円	暗号のすべて～ユビキタス社会の暗号技術	2,415円	〒310
通信技術とネットワーク	1,785円	〒310円	電子パーツ物知り百科	1,890円	〒310
新図解エレクトロ工作1	1,890円	〒310円	鉄道もの知り百科	1,890円	〒310

※上記以外にも新聞・出版物がございますので最寄りの本社、支局へお問い合わせ下さい。

東京本社	03-3445-8201(ダイヤルイン)	新潟支局	025-245-2526	京都支局	075-221-8021
大阪本社	06-6203-3361	関東総局	027-326-3206	広島支局	082-228-5581
西部本社	092-431-7411	松本支局	0263-36-2266	高松支局	087-861-3111
札幌支局	011-641-5591	名古屋支局	052-261-4541	鹿児島支局	099-226-3630
仙台支局	022-227-7211	金沢支局	076-263-8661		

Q10　携帯・PHSを持っていますか？
①両方持っている ②携帯 ③PHS ④購入予定 ⑤予定なし

Q11　Q10で④の方のみ，どちらを購入する予定ですか？
①携帯 ②PHS ③両方 ④未定

Q12　使用中のプログラミング言語は？
※枠内に一つご記入ください

Q13　これから勉強したいプログラミング言語は？
※枠内に一つご記入ください

Q14　今月号の本誌で面白かった記事を三つ，下の表からお答えください

Q15　今月号の本誌でつまらなかった記事を三つ，下の表からお答えください

ご協力ありがとうございました。

■今月号の記事一覧

① 編集部の学校訪問記…6
② 僕たち私たちのホームページへようこそ!…14
③ 記録型パソコン用DVD技術を知る…17
④ 今月の製品情報…22
⑤ マンガで学ぶ!コンピュータの世界!!…24
⑥ 大学入試センター試験対策直前マスター…28
⑦ パソコン認定試験対策講座…35
⑧ はじめようMicrosoft OfficeでBASICプログラミング…43
⑨ Visual Basicアルゴリズム講座…52
⑩ 続・ベーシックプログラミング教室…56
⑪ デジタルの世界へようこそ!…62
⑫ 特選投稿プログラム・コーナー…65
⑬ 今月のベスト・プログラマーは誰の手に！…109
⑭ Dr.Dのベスト・プログラムへの道…110
⑮ プログラムに関するQ&Aコーナー…112
⑯ PCハードウエアテクニック…114
⑰ HSPdeネットワークゲーム…120
⑱ .NETでゲームを作ろう！…124
⑲ PIECEプログラミング…132
⑳ ロボ道場…139
㉑ 自作一代記…143
㉒ あくせすROOM/プレゼント…147
㉓ SOFTWARE MAKER HOT INFORMATION…148
㉔ 表紙
㉕ OVER FLOW

マイコンBASIC Magazine　プログラム投稿応募カード

言語名（音源名）					
プログラム名（曲名）					
氏　名		ペンネーム		年齢	歳
住　所	〒				
TEL	※連絡可能な時間帯		応募年月日	年　月　日	

キリトリ線

郵便はがき

50円切手をお貼り下さい

141-8715

（受取人）

東京都品川区東五反田1−11−15

電波新聞社

月刊 マイコンBASICマガジン 編集部

2月号 行

(フリガナ)		年齢	性別
ご氏名		歳	男 ・ 女
□□□-□□□□		電話	
ご住所	都・道 府・県		

職業：小学生／中学生／高校生／高専生／短大生／大学生／専門学校生／予備校生／アルバイト／公務員／会社員／その他（　　　）

本誌を 毎月・時々・初めて 読む　何人で読みますか（　）人

自分のホームページを　持っている・持つ予定・持っていない・予定なし

本誌のほかによく読む雑誌は？［　　　］

ベーマガお楽しみプレゼント② 応募欄（147ページ掲載）	希望番号	品名
	番	

キリトリ線

月刊マイコンBASICマガジン② 愛読者カード

★OFコーナー宛 ♥あくせすROOM宛 ■質問・感想（本誌への掲載／可・不可）

ペンネーム：

■アンケート回答記入欄 ■ ※黒の筆記用具でわかりやすく，はっきりとご記入ください

Q1	□	Q2	□	〔メーカー名　〕	
Q3	□	Q4	□	Q5	□
Q6	□□□□□□□□				
Q7	□□□□□	〔商品名　〕			
Q8	□□□□□□□□				
Q9	□□□□□□□□				
Q10	□	Q11	□		
Q12	〔　〕	Q13	〔　〕		
Q14	□□□	Q15	□□□		

キリトリ線

■アンケートにご協力ください。

※左の愛読者カード・記入欄に，以下の質問に該当する番号をご記入ください。複数回答の質問で，回答欄をすべて埋める必要はありません。

Q1　メインで使っているパソコンは？
①デスクトップ ②ノート ③持っていない

Q2　パソコンの種類は？（Q1で③以外の方）
①自作 ②ショップブランド ③メーカー製
※枠内にメーカー名も記入してください

Q3　メインで使っているOSは？（Q1で③以外の方）
①Windows 95 ②Windows 98 ③Windows Me ④Windows 2000 ⑤Windows NT ⑥Windows XP ⑦Macintosh ⑧UNIX ⑨Linux ⑩その他

Q4　パソコン歴は？
①1年未満 ②1～2年 ③3～5年 ④6～10年 ⑤11年以上 ⑥なし

Q5　インターネット歴は？
①1年未満 ②1～2年 ③3～5年 ④6～10年 ⑤11年以上 ⑥なし

Q6　パソコンの主な使用目的は？（複数回答可）
①プログラミング ②ゲーム ③インターネット ④メール ⑤ミュージック ⑥学習 ⑦CG ⑧その他

Q7　これから買いたいと思うコンピュータ機器は？（複数回答可）
①パソコン本体 ②プリンタ ③スキャナ ④デジタルカメラ ⑤その他
※わかれば，商品名も記入してください

Q8　持っているゲーム機は？（複数回答可）
①プレイステーション（PSone含） ②プレイステーション2 ③ドリームキャスト ④ニンテンドウ64 ⑤ゲームキューブ ⑥Xbox ⑦ゲームボーイ ⑧ゲームボーイカラー ⑨ゲームボーイアドバンス ⑩ワンダースワン ⑪ワンダースワンカラー ⑫その他

Q9　これから買いたいと思うゲーム機は？（複数回答可）
①プレイステーション（PSone含） ②プレイステーション2 ③ドリームキャスト ④ニンテンドウ64 ⑤ゲームキューブ ⑥Xbox ⑦ゲームボーイカラー ⑧ワンダースワンカラー ⑨ゲームボーイアドバンス

※裏面へつづく→

■メモ欄…プログラム担当者・編集部へのご意見・ご感想など

ファルコムだから実現した、China英雄伝説！
武侠アクションRPGの決定版、ついに日本上陸！
「西風のラプソディー」に続くファルコム海外プロデュース第2弾！
武侠ARPG『月影のデスティニー』
Windows® 98/2000/Me/XP
2003年3月発売予定
月影のデスティニー
月影のDESTINY
http://www.falcom.co.jp/
■最新情報はファルコムホームページをご覧ください。 ■社員募集(新卒・既卒・中途)！詳しくはホームページ「会社案内・社員募集」をご覧ください。
Falcom®
NIHON FALCOM CORPORATION
〒190-0012 東京都立川市曙町1-14-13
☎042(527)6501 http://www.falcom.com/

第二十二巻第二号通巻二四八号 昭和五十七年十二月六日第三種郵便物認可 二〇〇三年一月二十五日印刷 二〇〇三年二月一日発行
電波新聞社 〒141-8715 東京都品川区東五反田一―十一―十五 電話〇三（三四四五）六一一一
定価550円 本体524円 雑誌18361-02 T1118361020557

2003

わかる！動かせる！
プログラムが組める雑誌

マイコンBASIC Magazine

2

■記録型パソコン用DVD技術を知る！

■大学入試センター試験・BASIC問題直前対策

http://www.basicmagazine.net/

電波新聞社